85-4248

DIGITAL MODEL FOR SIMULATING STEADY-STATE GROUND-WATER AND HEAT FLOW

U. S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 85-4248

NEWARK PUBLIC LIBRARY

JAN 14 1986

U. S. GOVT. PUBS.
DEPOSITORY 378

I 19.42/4:

DIGITAL MODEL FOR SIMULATING STEADY-STATE GROUND-WATER AND HEAT FLOW

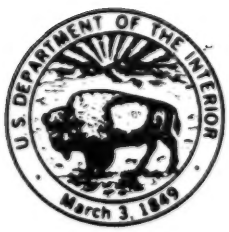
By J. E. Reed

U. S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 85-4248

Denver, Colorado

1985



UNITED STATES DEPARTMENT OF THE INTERIOR

DONALD PAUL HODEL, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information write to: M.S. Bedinger, Project Chief, Task I Project, Box 25046, MS 417, Federal Center, Denver, CO 80225

(303) 236-4973

Copies of this report can be purchased from: Western Distribution Branch, U.S. Geological Survey, Box 25425, Federal Center, Denver, CO 80225

(303) 236-7477

CONTENTS

	Page
Abstract	1
Introduction	2
Ground-water-flow model	9
Heat-flow model	16
Solution procedure	25
Travel time	30
Computer program	33
References	36
Attachment 1, Program listing	38
Attachment 2, Definition of selected program variables	102
Attachment 3, Input data	108
Attachment 4, Description of output data files	114
Attachment 5, Example simulation	116
Attachment 6, Generalized flow chart for program	127

ILLUSTRATIONS

Figure 1.	Diagram showing three types of finite-difference grids	7
2.	Sketch of finite-difference block for three-dimensional ground-water flow	8
3.	Sketch illustrating equation numbering for a 4 by 4 by 4 grid	26
4.	Diagram illustrating structure of coefficient matrix for a 4 by 4 by 4 grid	27

CONVERSION FACTORS

Multiply unit	By	To obtain metric unit
inch (in.)	25.4	millimeter (mm)
foot (ft)	0.3048	meter (m)
inch per year (in./yr)	$8.049*10^{-7}$	millimeter per second (mm/s)
cubic foot per day (ft^3/d)	$3.277*10^{-7}$	cubic meter per second (m^3/s)
degree Fahrenheit (^oF) - 32	0.5555	degree Celsius (^oC)
British thermal unit per day (BTU/d)	12.20	milliwatts (mW)
British thermal unit per cubic foot per degree Fahrenheit ($BTU/ft^3 - F^o$)	67.02	joule per liter per degree Celsius ($J/L- C^o$)
British thermal unit per day per square foot per degree Fahrenheit ($BTU/d-ft^2 - F^o$)	236.4	milliwatts per square meter per degree Celsius ($mW/m^2 - C^o$)
heat flow unit (HFU); 10^{-6} calories per square centimeter per second	41.84	milliwatt per square meter (mW/m^2)
conductance unit (CU); 10^{-3} calories per centimeter per second per degree Celsius	418.4	milliwatt per meter per degree Cesius ($mW/m- C^o$)

DIGITAL MODEL FOR SIMULATING STEADY-STATE GROUND-WATER AND HEAT FLOW

By J. E. Reed

ABSTRACT

The computer program models steady-state water and heat flow in an isotropic, heterogeneous, three-dimensional aquifer system with uniform thermal properties and no change of state. Driving forces on the system are external hydrologic conditions of recharge from precipitation and fixed hydraulic-head boundaries. Heat flux includes geothermal heat-flow, conduction to the land surface, advection (heat conveyed by water) from recharge, and advection to or from fixed-head boundaries.

The program uses an iterative procedure that alternately solves the ground-water-flow and heat-flow equations, updating advective flux after solution of the ground-water-flow equation, and updating hydraulic conductivity after solution of the heat-flow equation. Direct solution is used for each equation.

Time of travel is determined by particle tracking through the modeled space. Velocities within blocks are linear interpolations of velocities at block faces.

Models of cross-sections display additional information along selected flow paths and for lithology within the section.

INTRODUCTION

This report documents the techniques used in the program HOTWTR to model the coupled, three-dimensional, steady-state flow of water and heat through isotropic heterogeneous porous media. This program was developed as an aid in screening large ground-water-flow systems as prospects for underground waste storage. The structure and content of the program reflect that task and the program is not intended to be a general purpose approach to a considerable range of ground-water problems.

Most of the techniques are variations of commonly accepted modeling practices. Sources for program concepts include Trescott, Pinder, and Larson (1976), Konikow and Bredehoeft (1978),and Faust and Mercer (1977). An earlier version of this model is discussed in Bedinger and others (1979). Only an outline of the mathematical development is included in this report. Discussion of alternative methods is included in places to justify the procedure selected. Readers unfamiliar with the concepts of finite-difference methods will find a comprehensive discussion of these methods in Bennett (1976, p. 119).

The program is based on separate finite-difference approximations of the ground-water-flow equation and of the convective-dispersive heat-flow equation. The ground-water-flow and heat-flow models are interdependent because hydraulic conductivity in the ground-water-flow model is a function of temperature and convective flow in the heat-flow model is a function of ground-water flow. The program iterates on the coupling between the two equations. The water- and heat-flow equations are solved alternately, with water flux through the block faces updated after solution of the ground-water-flow equation, and hydraulic conductivity updated after solution of the heat-flow equation. The variation of hydraulic conductivity with temperature is principally due to the change in viscosity, and, therefore, density can be assumed to be sensibly constant. The computer program also may be used as an isothermal model in which only the ground-water flow equation solved.

The program allows the hydraulic conductivity and porosity of the rock to be discretized in space, but the thermal conductivity of the fluid-saturated rock and the specific heat capacity of the fluid need to be assigned constant values. Also, for the program, the fluid is assumed to be of constant density. Hence, the program cannot be used to model free convection, which is one of two types of convection in porous media identified by Sorey (1978, p.D8). Forced convection results from stresses on the ground-water system. Free convection results from buoyancy effects caused by fluid-density variations due to temperature differences. Geothermal systems contain both types of convection (Sorey, p.D9). This model only considers forced convection, and is not applicable to systems where free convection is significant as indicated by geysers, steam, large temperature gradients or substantial heat flow.

Necessary data for this program consist of the geometric distribution of lithologies within the modeled space, the hydraulic conductivity and porosity for each lithologic type, the water level at recharge or discharge sites such as lakes or streams, recharge from infiltration of precipitation, temperature of recharged water, land-surface temperature, geothermal heat-flow, average thermal conductivity of the water-saturated rock, and volumetric specific heat of the fluid.

The ground-water-flow and heat-flow models are block-centered, finite-difference approximations with interblock hydraulic transmissivity calculated as the harmonic mean of the two adjacent blocks.

The finite-difference method considers the space of interest to be represented by a finite number of blocks. Variable properties of the flow system, such as hydraulic conductivity, are replaced in the model by their average value within each block. The average value for the variable to be calculated, hydraulic head or temperature, is assigned to a position called the node. Each node is enclosed by a subdomain of the flow system.

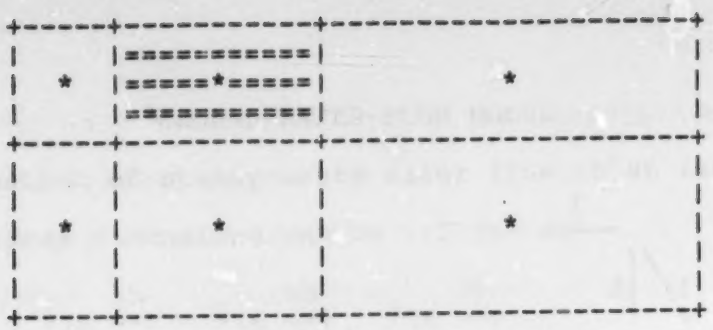
In a block-centered grid, the node is always centered between adjacent block faces and each flow subdomain is identical to a block. For point-distributed grids, each subdomain face is centered between adjacent nodes. Point-distributed grids with each subdomain identical to a block also are called face-centered grids.

For grids having uniform spacing in all directions, block-centered and face-centered grids are equivalent. If block size (therefore node spacing) is variable, the nodes in a face-centered grid will not be in the center of all blocks. In general, a face-centered grid with a minimum number of nodes cannot be designed because the block faces will not always coincide with natural boundaries in the modeled space. However, a block-centered grid can always be designed to match natural boundaries.

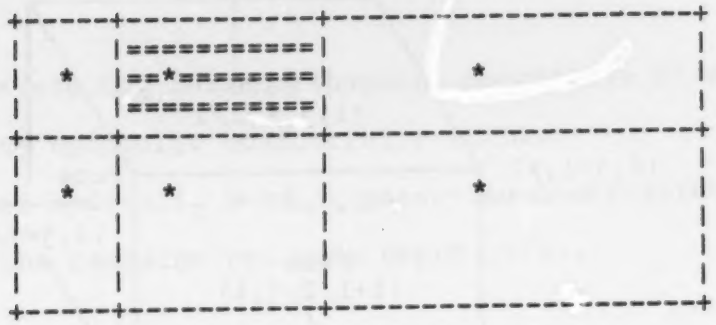
Point-distributed grids can be designed that match natural boundaries (R. L. Cooley, U.S. Geological Survey, oral commun., 1985). Such grids have nodes at block corners and flow subdomains include parts of four adjacent blocks. Equations developed for a point-distributed grid are a better approximation than are equations developed for a block-centered grid. However, velocity distribution in the flow subdomain could be more complex if the subdomain includes blocks of differing hydraulic conductivity or porosity. Examples of these grids are illustrated in Figure 1.

Subscript notation is used here to reference a quantity's position in the finite-difference grid. Integer subscripts refer to blocks or to nodes within blocks, whereas fractional-half subscripts refer to block faces. For example, in reference to node i, j, and k, where i, j, and k are integer numbers, the head (h) at node i, j, k is denoted by $h_{i,j,k}$, the head at the adjacent node in the i increasing direction is denoted by $h_{i+1,j,k}$, the specific discharge, q, at the block face in the i increasing direction is denoted by $q_{i+1/2,j,k}$, and so forth. The coordinate axes of a finite-difference block are illustrated in Figure 2.

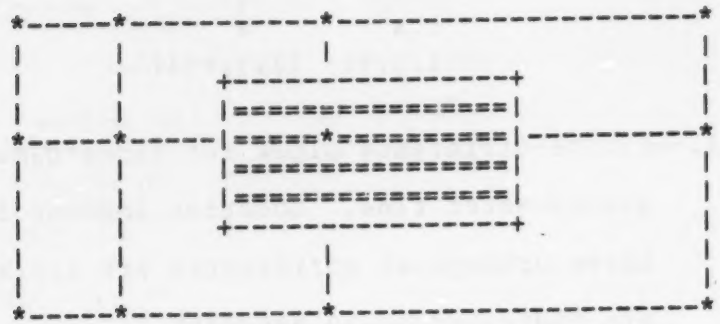
Dimensions for coefficients in equations are given in general units, not in specific units. These abbreviations (and their units) are L (length), T (time), E (energy), and Deg. (temperature). Energy is used here as a general unit although energy has dimensions of M*L**2/T**2.



Block-centered grid



Face-centered grid



Cooley's point-distributed grid

Explanation

* Nodes

, --- Block boundaries

One selected flow subdomain

Figure 1.-Three types of finite-difference grids.

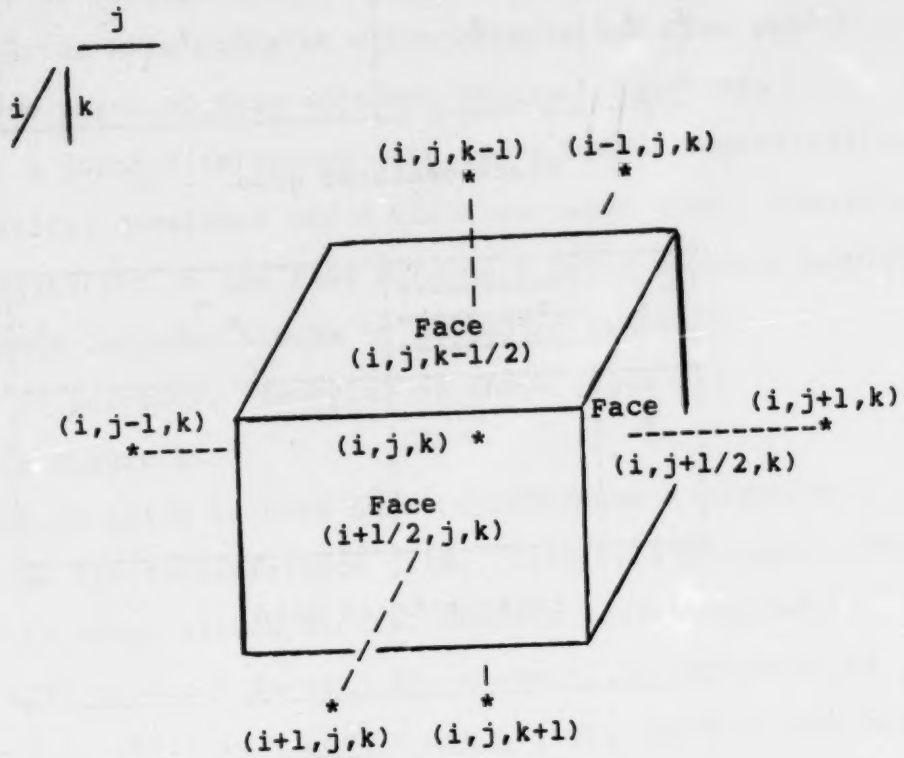


Figure 2.--Finite-difference block for three-dimensional ground-water flow. Location indices for the three orthogonal corrdinates are i,j.k. Nodes are indicated by an asterisk.

GROUND-WATER-FLOW MODEL

The equation of steady-state water flow in an isotropic aquifer for three dimensions can be written as

$$\frac{\partial\left(K\frac{\partial h}{\partial y}\right)}{\partial y} + \frac{\partial\left(K\frac{\partial h}{\partial x}\right)}{\partial x} + \frac{\partial\left(K\frac{\partial h}{\partial z}\right)}{\partial z} + Ws = 0 \qquad (1)$$

where

y, x, z are the three orthogonal coordinate directions (L),

K is the hydraulic conductivity (L/T),

h is the hydraulic head or water-level elevation (L), and

Ws is the recharge per unit depth (1/T).

A finite-difference approximation to equation 1 can be written as

$$\frac{K_{i+1/2,j,k}\,(h_{i+1,j,k} - h_{i,j,k})}{(Dy_{i+1} + Dy_i)/2} - \frac{K_{i-1/2,j,k}\,(h_{i,j,k} - h_{i-1,j,k})}{(Dy_i + Dy_{i-1})/2} \Bigg/ Dy_i$$

$$+ \frac{K_{i,j+1/2,k}\,(h_{i,j+1,k} - h_{i,j,k})}{(Dx_{j+1} + Dx_j)/2} - \frac{K_{i,j-1/2,k}\,(h_{i,j,k} - h_{i,j-1,k})}{(Dx_j + Dx_{j-1})/2} \Bigg/ Dx_j$$

$$+ \frac{K_{i,j,k+1/2}\,(h_{i,j,k+1} - h_{i,j,k})}{(Dz_{k+1} + Dz_k)/2} - \frac{K_{i,j,k-1/2}\,(h_{i,j,k} - h_{i,j,k-1})}{(Dz_k + Dz_{k-1})/2} \Bigg/ Dz_k \qquad (2)$$

$$+ Ws_{i,j,k} = 0$$

In equation 2, the subscripts represent position in the finite-difference grid,

i,j,k indicates a specific node or block,

i-1 and i+1 indicate the preceding and succeeding nodes or blocks in the y direction,

j-1,j+1, k-1, and k+1 are likewise for the x and z directions respectively,

i-1/2 and i+1/2 indicate preceding and succeeding block faces in the y direction,

j-1/2, j+1/2, k-1/2, and k+1/2 indicate likewise for the x and z directions, respectively,

Dy, Dx, and Dz are block lengths in the y,x, and z directions, respectively,

and other symbols are as defined previously. The above finite-difference approximation applied to a region of interest results in a system of n (number of blocks) linear equations for n unknown hydraulic heads.

The hydraulic conductivity at the block face is the thickness-weighted harmonic mean of the hydraulic conductivities for the adjacent blocks. For the y direction this is

$$K_{i-1/2,j,k} = \frac{K_{i-1,j,k} K_{i,j,k} (Dy_i + Dy_{i-1})}{K_{i-1,j,k} Dy_i + K_{i,j,k} Dy_{i-1}}$$

and

$$K_{i+1/2,j,k} = \frac{K_{i+1,j,k} K_{i,j,k} (Dy_i + Dy_{i+1})}{K_{i+1,j,k} Dy_i + K_{i,j,k} Dy_{i+1}}$$

Similar expressions may be developed for the hydraulic conductivity at block faces in the x and z directions.

The finite-difference equation for a block multiplied by the volume of the block ($Dy_i * Dx_j * Dz_k$) becomes a balance equation with each term representing a flow into or out of the block. The balance equation requires that total inflow to the block must equal total outflow. Equation 2 becomes

$$\left[\frac{K_{i+1/2,j,k}\,(h_{i+1,j,k} - h_{i,j,k})}{(Dy_{i+1} + Dy_i)/2} - \frac{K_{i-1/2,j,k}\,(h_{i,j,k} - h_{i-1,j,k})}{(Dy_i + Dy_{i-1})/2}\right] Dx_j\, Dz_k$$

$$+\left[\frac{K_{i,j+1/2,k}\,(h_{i,j+1,k} - h_{i,j,k})}{(Dx_{j+1} + Dx_j)/2} - \frac{K_{i,j-1/2,k}\,(h_{i,j,k} - h_{i,j-1,k})}{(Dx_j + Dx_{j-1})/2}\right] Dy_i\, Dz_k$$

$$+\left[\frac{K_{i,j,k+1/2}\,(h_{i,j,k+1} - h_{i,j,k})}{(Dz_{k+1} + Dz_k)/2} - \frac{K_{i,j,k-1/2}\,(h_{i,j,k} - h_{i,j,k-1})}{(Dz_k + Dz_{k-1})/2}\right] Dy_i\, Dx_j$$

$$+ Ws_{i,j,k}\, Dy_i\, Dx_j\, Dz_k = 0 \qquad (3)$$

where all terms are as previously defined.

Collecting terms in equation 3 gives the form

$$A_{i,j,k}\, h_{i-1,j,k} + B_{i,j,k}\, h_{i,j-1,k} + C_{i,j,k}\, h_{i,j,k-1} + D_{i,j,k}\, h_{i,j,k}$$

$$+ E_{i,j,k}\, h_{i,j,k+1} + F_{i,j,k}\, h_{i,j+1,k} + G_{i,j,k}\, h_{i+1,j,k}$$

$$+ Ws_{i,j,k}\, Dy_i\, Dx_j\, Dz_k = 0. \qquad (4)$$

The terms in equation 4 are:

$$A_{i,j,k} = \frac{K_{i-1/2,j,k}\, Dx_j\, Dz_k}{(Dy_i + Dy_{i-1})/2},$$

$$B_{i,j,k} = \frac{K_{i,j-1/2,k}\, Dy_i\, Dz_k}{(Dx_j + Dx_{j-1})/2},$$

$$C_{i,j,k} = \frac{K_{i,j,k-1/2}\, Dy_i\, Dx_j}{(Dz_k + Dz_{k-1})/2},$$

$$E_{i,j,k} = \frac{K_{i,j,k+1/2}\, Dy_i\, Dx_j}{(Dz_{k+1} + Dz_k)/2},$$

$$F_{i,j,k} = \frac{K_{i,j+1/2,k}\, Dy_i\, Dz_k}{(Dx_{j+1} + Dx_j)/2},$$

$$G_{i,j,k} = \frac{K_{i+1/2,j,k}\, Dx_j\, Dz_k}{(Dy_{i+1} + Dy_i)/2},$$

and

$$D_{i,j,k} = -(A_{i,j,k} + B_{i,j,k} + C_{i,j,k} + E_{i,j,k} + F_{i,j,k} + G_{i,j,k}).$$

Equation 4 can be written for each of the n active blocks in the model resulting in a system of n equations for n unknown hydraulic heads. This system, in matrix notation, is

$$[B]\{h\}=\{-(W+H)\}$$

where B is the matrix of coefficients of the hydraulic heads in the system of equations, h is the vector (n by 1 matrix) of unknown hydraulic heads, -(W+H) is a vector of known terms, formed by moving the recharge and constant hydraulic-head terms to the right side of the equations.

In residual form (Trescott and others, 1976, p.16, eq. 17d) the system becomes

$$[B]\{hi+Dh\}=\{-(W+H)\}$$

where h is separated into hi, the initial hydraulic head, and Dh, the computed change in hydraulic head. The residual form decreases rounding errors in the computation when change in hydraulic head is smaller in magnitude than hydraulic head. Transfering the known terms to the right side gives

$$[B]\{Dh\}=-([B]\{hi\}+\{W+H\})$$

or

$$[B]\{Dh\}=\{R\}$$

where R is a vector composed of all known terms.

HEAT-FLOW MODEL

The equation for steady-state, convective-diffusive heat flow in three dimensions for conditions of uniform thermal conductivity and specific heat may be written

$$Kt\left(\frac{\partial^2 t}{\partial y^2} + \frac{\partial^2 t}{\partial x^2} + \frac{\partial^2 t}{\partial z^2}\right) + St\left[\frac{\partial (qt)}{\partial y} + \frac{\partial (qt)}{\partial x} + \frac{\partial (qt)}{\partial z}\right] + Ht = 0 \qquad (5)$$

where

Kt is the thermal conductivity of the rock saturated with water, [E/(L*T*Deg.)],

t is the temperature (degrees on some scale),

y, x, z are the three orthogonal coordinates (L),

St is the volumetric specific heat for the fluid [E/(L**3*Deg.)],

q is the specific discharge, flow per unit area, of ground water at a point (L/T),

and Ht is a heat source term which includes the geothermal heat flow per unit depth [E/(L**3*T)].

A finite-difference approximation of equation 5 can be written for a block numbered i,j,k as

$$
\begin{aligned}
&Kt\left[\frac{\dfrac{t_{i+1,j,k}-t_{i,j,k}}{(Dy_{i+1}+Dy_i)/2}-\dfrac{t_{i,j,k}-t_{i-1,j,k}}{(Dy_i+Dy_{i-1})/2}}{Dy_i}\right.\\
&+\frac{\dfrac{t_{i,j+1,k}-t_{i,j,k}}{(Dx_{j+1}+Dx_j)/2}-\dfrac{t_{i,j,k}-t_{i,j-1,k}}{(Dx_j+Dx_{j-1})/2}}{Dx_j}\\
&\left.+\frac{\dfrac{t_{i,j,k+1}-t_{i,j,k}}{(Dz_{k+1}+Dz_k)/2}-\dfrac{t_{i,j,k}-t_{i,j,k-1}}{(Dz_k+Dz_{k-1})/2}}{Dz_k}\right]\\
&+St\left(\frac{q_{i+1/2,j,k}\,t_{i+1/2,j,k}-q_{i-1/2,j,k}\,t_{i-1/2,j,k}}{Dy_i}\right.\\
&+\frac{q_{i,j+1/2,k}\,t_{i,j+1/2,k}-q_{i,j-1/2,k}\,t_{i,j-1/2,k}}{Dx_j}\\
&\left.+\frac{q_{i,j,k+1/2}\,t_{i,j,k+1/2}-q_{i,j,k-1/2}\,t_{i,j,k-1/2}}{Dz_k}\right)+Ht_{i,j,k}=0. \qquad (6)
\end{aligned}
$$

In equation 6, the subscript notation is as discussed previously and t, Kt, St, q, and Ht are as defined above.

The specific discharge, q, is calculated for the y direction as

$$q_{i-1/2,j,k} = \frac{K_{i-1/2,j,k}(h_{i,j,k} - h_{i-1,j,k})}{(Dy_i + Dy_{i-1})/2}$$

and

$$q_{i+1/2,j,k} = \frac{K_{i+1/2,j,k}(h_{i,j,k} - h_{i+1,j,k})}{(Dy_i + Dy_{i+1})/2} .$$

Similar expressions may be developed for the specific discharge at block faces in the x and z directions.

The effective temperature for convective flow by upgradient weighting is

$$t_{i+1/2,j,k} = t_{i,j,k} \quad \text{if } q_{i+1/2,j,k} > 0$$

or

$$t_{i+1/2,j,k} = t_{i+1,j,k} \quad \text{if } q_{i+1/2,j,k} < 0$$

and

$$t_{i-1/2,j,k} = t_{i-1,j,k} \quad \text{if } q_{i-1/2,j,k} > 0$$

or

$$t_{i-1/2,j,k} = t_{i,j,k} \quad \text{if } q_{i-1/2,j,k} < 0$$

and similarly for the other two directions. Upgradient weighting for convective heat transport assigns face temperature to the "upgradient" node. For central weighting, face temperature is computed as an average for the two adjacent nodes. Central weighting is more accurate for small grid spacing than upgradient weighting, but produces oscillatory, unstable solutions when the grid spacing is large. Upgradient weighting provides a stable solution for large grid spacing.

Multiplying the finite-difference equation, equation 6, by the volume of the block results in a balance equation for heat flow which is:

$$
\begin{aligned}
Kt[\,&\left(\frac{t_{i+1,j,k} - t_{i,j,k}}{(Dy_{i+1} + Dy_{i})/2} - \frac{t_{i,j,k} - t_{i-1,j,k}}{(Dy_{i} + Dy_{i-1})/2}\right) Dx_{j}\, Dz_{k} \\
&+ \left(\frac{t_{i,j+1,k} - t_{i,j,k}}{(Dx_{j+1} + Dx_{j})/2} - \frac{t_{i,j,k} - t_{i,j-1,k}}{(Dx_{j} + Dx_{j-1})/2}\right) Dy_{i}\, Dz_{k} \\
&+ \left(\frac{t_{i,j,k+1} - t_{i,j,k}}{(Dz_{k+1} + Dz_{k})/2} - \frac{t_{i,j,k} - t_{i,j,k-1}}{(Dz_{k} + Dz_{k-1})/2}\right) Dx_{j}\, Dy_{i}\,] \\
+ St[\,&(q_{i+1/2,j,k}\, t_{i+1/2,j,k} - q_{i-1/2,j,k}\, t_{i-1/2,j,k})\, Dx_{j}\, Dz_{k} \\
&+ (q_{i,j+1/2,k}\, t_{i,j+1/2,k} - q_{i,j-1/2,k}\, t_{i,j-1/2,k})\, Dy_{i}\, Dz_{k} \\
&+ (q_{i,j,k+1/2}\, t_{i,j,k+1/2} - q_{i,j,k-1/2}\, t_{i,j,k-1/2})\, Dy_{i}\, Dx_{j}\,] \\
&+ Ht_{i,j,k}\, Dy_{i}\, Dx_{j}\, Dz_{k} = 0.
\end{aligned}
\tag{7}
$$

Each term in the preceding balance equation is a heat flow (dimensions E/T) into or out of the block, with total inflow equal to total outflow.

Collecting terms in equation 7 gives the form

$$A_{i,j,k}\, t_{i-1,j,k} + B_{i,j,k}\, t_{i,j-1,k} + C_{i,j,k}\, t_{i,j,k-1} + D_{i,j,k}\, t_{i,j,k}$$

$$+ E_{i,j,k}\, t_{i,j,k+1} + F_{i,j,k}\, t_{i,j+1,k} + G_{i,j,k}\, t_{i+1,j,k}$$

$$+ Ht_{i,j,k}\, Dy_i\, Dx_j\, Dz_k = 0 \qquad (8)$$

with

$$A_{i,j,k} = \frac{Kt\, Dx_j\, Dz_k}{(Dy_i + Dy_{i-1})/2} - St\, q_{i-1/2,j,k}\, Dx_j\, Dz_k$$

if $q_{i-1/2,j,k} > 0$

or

$$A_{i,j,k} = \frac{Kt\, Dx_j\, Dz_k}{(Dy_i + Dy_{i-1})/2}$$

if $q_{i-1/2,j,k} < 0$

and

$$B_{i,j,k} = \frac{Kt\, Dy_i\, Dz_k}{(Dx_j + Dx_{j-1})/2} - St\, q_{i,j-1/2,k}\, Dy_i\, Dz_k$$

if $q_{i,j-1/2,k} > 0$

or

$$B_{i,j,k} = \frac{Kt\, Dy_i\, Dz_k}{(Dx_j + Dx_{j-1})/2}$$

if $q_{i,j-1/2,k} < 0.$

Also,

$$C_{i,j,k} = \frac{Kt\ Dy_i\ Dx_j}{(Dz_k + Dz_{k-1})/2} - St\ q_{i,j,k-1/2}\ Dy_i\ Dx_j$$

if $q_{i,j,k-1/2} > 0$

or

$$C_{i,j,k} = \frac{Kt\ Dy_i\ Dx_j}{(Dz_k + Dz_{k-1})/2}$$

if $q_{i,j,k-1/2} < 0$

and

$$E_{i,j,k} = \frac{Kt\ Dy_i\ Dx_j}{(Dz_k + Dz_{k+1})/2}$$

if $q_{i,j,k+1/2} > 0$

or

$$E_{i,j,k} = \frac{Kt\ Dy_i\ Dx_j}{(Dz_k + Dz_{k+1})/2} - St\ q_{i,j,k+1/2}\ Dy_i\ Dx_j$$

if $q_{i,j,k+1/2} < 0$.

Also,

$$F_{i,j,k} = \frac{Kt\ Dy\ Dz}{(Dx_j + Dx_{j+1})/2}$$

if $q_{i,j+1/2,k} > 0$

or

$$F_{i,j,k} = \frac{Kt\ Dy\ Dz}{(Dx_j + Dx_{j+1})/2} - St\ q_{i,j+1/2,k}\ Dy\ Dz$$

if $q_{i,j+1/2,k} < 0$

and

$$G_{i,j,k} = \frac{Kt\ Dx\ Dz}{(Dy_i + Dy_{i+1})/2}$$

if $q_{i+1/2,j,k} > 0$

or

$$G_{i,j,k} = \frac{Kt\ Dx\ Dz}{(Dy_i + Dy_{i+1})/2} - St\ q_{i+1/2,j,k}\ Dx\ Dz$$

if $q_{i+1/2,j,k} < 0$

and

$D_{i,j,k}$ =-(sum of the alternate terms for $A_{i,j,k}$, $B_{i,j,k}$, $C_{i,j,k}$, $E_{i,j,k}$, $F_{i,j,k}$, and $G_{i,j,k}$ with sign of q terms reversed if present).

Equation 8 can be written for each of the n active blocks in the model, resulting in a system of n equations for n unknown temperatures.

This system, in matrix notation, is

$$[B]\{t\}=\{-(H'+T')\}$$

where B is the matrix of coefficients of the temperatures in the system of equations, t is the vector of unknown temperatures, -(H'+T') is a vector of known terms, formed by moving the heat flow, H', and constant temperature, T', terms to the right side of the equations. This system in residual form is

$$[B]\{Dt\}=-([B]\{ti\}+\{H'+T'\})$$

or

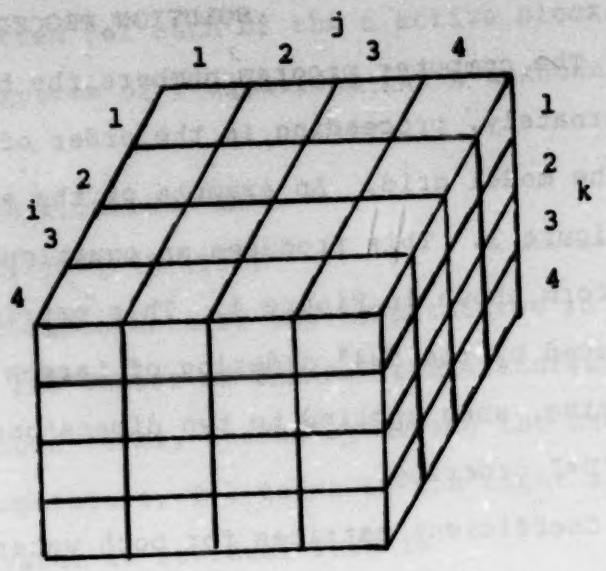
$$[B]\{Dt\}=\{R\}$$

where Dt is the change in temperature, ti is the initial temperature, and R is the vector of known terms.

SOLUTION PROCEDURE

The computer program numbers the balance equations alternately, proceeding in the order of the smallest dimensions of the model grid. An example of the equation numbering is shown in Figure 3. This produces an equation coefficent matrix with the form shown in Figure 4. This matrix is similar to that produced by the "D4" ordering of Larson (1978). However, the ordering, when applied to two dimensions, is less efficient than the "D4" ordering.

Coefficient matrices for both water and heat balances have the form shown in figure 4. Of course, the coefficients have different values. In particular, the matrix for water balance is symmetric, whereas the matrix for heat balance is not. The same procedure is used to solve both.



j

i \	1	2	3	4	
1	1	35	5	39	k=1
2	41	11	45	15	
3	17	51	21	55	
4	57	27	61	31	

i \	1	2	3	4	
1	33	3	37	7	k=2
2	9	43	13	47	
3	49	19	53	23	
4	25	59	29	63	

i \	1	2	3	4	
1	2	36	6	40	k=3
2	42	12	46	16	
3	18	52	22	56	
4	58	28	62	32	

i \	1	2	3	4	
1	34	4	38	8	k=4
2	10	44	14	48	
3	50	20	54	24	
4	26	60	30	64	

Figure 3.--Equation numbering for a 4 by 4 by by 4 grid

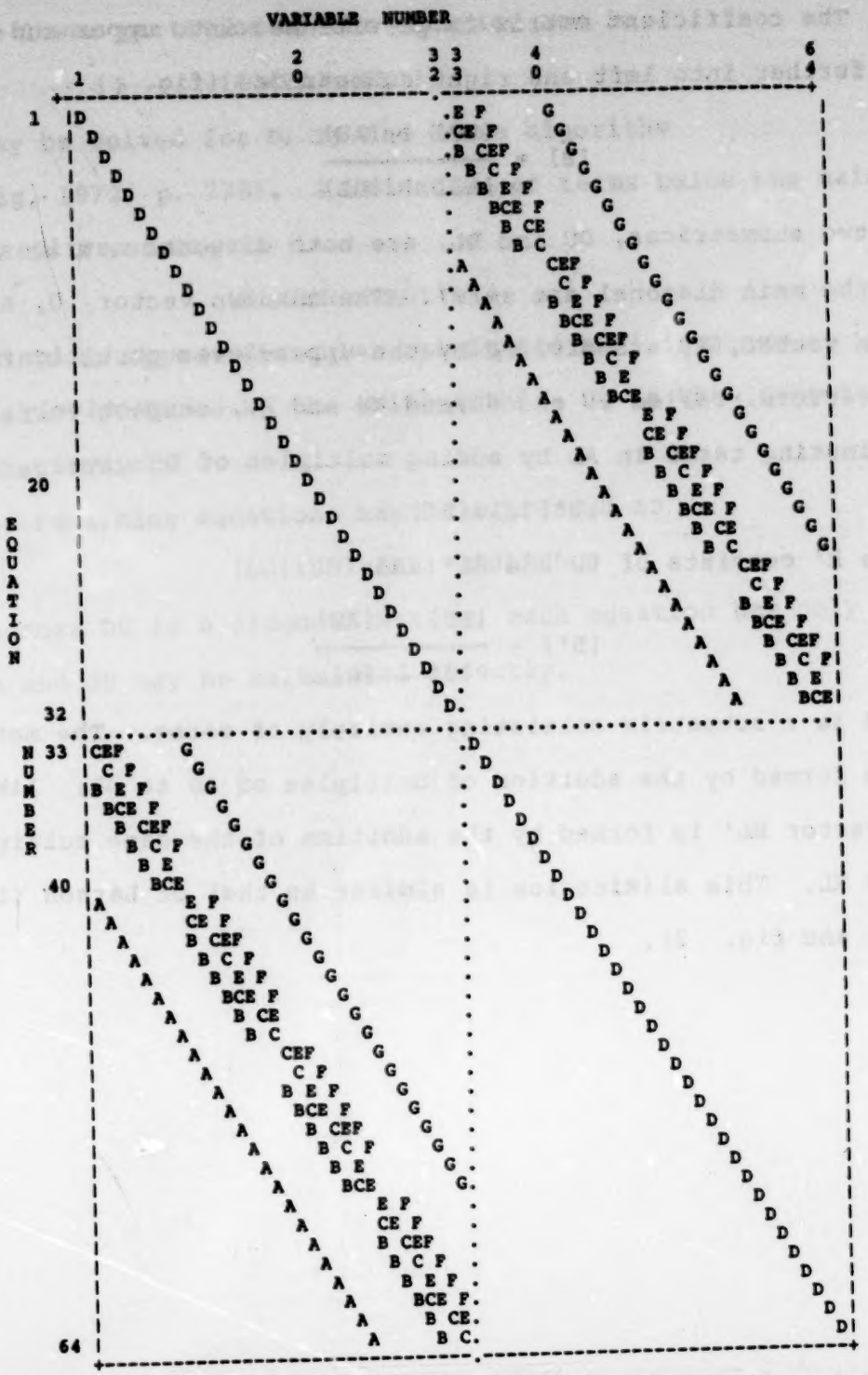


Figure 4.--Structure of coefficient matrix for a 4 by 4 by 4 grid. Symbols A,B,C,D,E,F, and G refer to coefficients in equations 4 and 8. Blanks indicate zero coefficients. Dotted lines indicate the partitioning of the matrix.

The coefficient matrix is partitioned into upper and lower, and further into left and right submatrices (fig. 4)

$$[B] = \frac{[DU] \mid [AU]}{[AL] \mid [DL]}$$

The two submatrices, DU and DL, are both diagonal matrices (terms off the main diagonal are zero). The unknown vector, U, and known vector, R, are divided by the upper-lower partitioning into two vectors, called UU and UL, and RU and RL, respectively. Eliminating terms in AL by adding multiples of DU gives

$$[B']\{U\}=\{R'\}$$

where R' consists of RU and RL' and

$$[B'] = \frac{[DU] \mid [AU]}{[0] \mid [A]}$$

and 0 is a submatrix consisting entirely of zeros. The matrix, A, is formed by the addition of multiples of AU to DL; likewise, the vector RL' is formed by the addition of the same multiples of RU to RL. This elimination is similar to that of Larson (1978, p. 3 and fig. 2).

The lower partition of the equation now is

$$[A]\{UL\}=\{RL'\}$$

which may be solved for UL by the Gauss algorithm (Kreyszig, 1972, p. 235). Elimination of terms below the main diagonal in A results in

$$[A']\{UL\}=\{RL''\}$$

where A' is an upper triangular matrix and RL'' is a new known-vector produced by the elimination. UL is then computed by back substitution.

The remaining equations may be expressed as

$$[DU]\{UU\}=\{RU\}-[AU]*UL$$

and, because DU is a diagonal matrix, each equation has only one unknown and UU may be calculated directly.

TRAVEL TIME

Travel time is computed in the code by moving a particle of water through the model. The velocity at a block face is

$$v_{i-1/2,j,k} = \frac{q_{i-1/2,j,k}}{p_{i,j,k}}$$

and

$$v_{i+1/2,j,k} = \frac{q_{i+1/2,j,k}}{p_{i,j,k}}$$

and similarly for the other two directions, where the subscripts indicate location as defined previously,

v is the velocity (L/T),

q is the specific discharge (L/T), and

p is the porosity (dimensionless).

The velocity in a given direction within a block is estimated as a linear interpolation between the velocities at the two opposing faces. For example,

$$vy = ay+by*y$$

where vy is the velocity in the y direction, and ay and by are constants defining the linear function. Using a local coordinate system for each block with y =0 at face i-1/2,j,k and y = Dy_i at face i+1/2,j,k the two constants are

$$ay = v_{i-1/2,j,k}$$

and

$$by = \frac{v_{i+1/2,j,k} - v_{i-1/2,j,k}}{Dy_i}$$

Velocities are computed in a similar manner for the other two directions.

The time, $Dty_{i,j,k}$, that it takes to travel from a point within a block, yo, to a point on a block face, yf, along the component of movement in the y direction is given by

$$Dty_{i,j,k} = \int_{yo}^{vf} \frac{1}{ay + by*y}\, dy$$

where yo is the lower limit of integration, yf is the upper limit, 1/(ay+by*y) is the integrand, dy is the differential of y, and ay+by*y is the linear velocity function. Performing the integration

$$Dty_{i,j,k} = \frac{1}{by} \ln \frac{ay + by*yf}{ay + by*yo} \qquad (9)$$

where ln is the natural (base e) logarithm. Similar development results in similar expressions for the x and z directions.

The exit point (xf,yf, zf) from the block is unknown. Substituting the maximum possible values for xf,yf, and xf into equation 9 results, in general, in three different travel times. The flow path intersects the face in the direction of the coordinate associated with the minimum time. The other two coordinates of the exit point may be determined by substituting the minimum time into equation 9.

The travel time across a block is the minimum of the travel times obtained by applying eq. 9 in the three coordinate directions. The total travel time along a flow path through the model is the sum of the travel times through the blocks traversed by the flow path.

COMPUTER PROGRAM

The program, written in Fortran 77, consists of the main routine, HOTWTR, and 27 subroutines. The code is listed in attachment 1. Definition of selected program variables is in attachment 2. Description of the input data is given in attachment 3. An outline of the output data is given in attachment 4. An example simulation showing input data and selected parts of the PRINT output data is in attachment 5. A generalized flow chart for the program is in attachment 6. All attachments are at the back of the report. A tabulation of the subroutines follows:

Name	Called from	Description
BACK	HOTWTR	Back substitutes to determine upper part of unknowns, adds change to old value, and determines largest change.
BALM	HOTWTR	Computes heat and water balance.
COMPQ	HOTWTR	Computes interblock ground-water flow.
DATAIN	HOTWTR	Reads input data.
DREAD	DATAIN	Reads double-precision arrays.
HEATEQ	HOTWTR	Computes coefficient matrix and known vector for the heat equation.
MOVE	HOTWTR	Determines time of travel and point of discharge for each active node.
MSUB	MOVE and PATH	Tracks water particle through the model.
MSUB1	MSUB	Calculates position, velocity, and travel time to block face in direction of movement.

Name	Called from	Description
MSUB2	MSUB	Determines new coordinates.
NUMBER	DATAIN	Numbers equations and sets pointers from model grid to equation.
PA	HOTWTR	Adjusts hydraulic conductivity for temperature change.
PARTN	HOTWTR	Sets pointers from equation to model grid.
PATH	HOTWTR	Traces flow paths from top of model to discharge point.
PLINE1	PRINT	Constructs one line of flow-path cross-section.
PLINE2	PRINT	Constructs one line of head or travel-time cross-section.
PLINE4	PRINT	Constructs one line of lithology cross-section.
PRINT	HOTWTR	Constructs cross-sections of flow paths, head, travel time, and lithology.
PSUB	MSUB	Saves flow-path coordinates.
READ	DATAIN	Reads single-precision arrays.
SCALEH	PRINT	Scales head values.
SCALET	PRINT	Scales time of travel.
SOLVE	HOTWTR	Reduces the A matrix to A' and back substitutes to determine the lower part of the unknowns.
SORT	HOTWTR	Sorts flow-path coordinates.
VELO	HOTWTR	Computes velocities.
WATEQ	HOTWTR	Computes the coefficient matrix and the known vector for the ground-water equations.
ZER	MSUB2 and PATH	Locates limiting points near points of zero velocity.

Size of the A array (253000) permits a 60*60 two-dimensional or a 12*12*12 three-dimensional model. Storage requirements for the A array can be estimated as (mn*2+1)*(mnl/2) where mn is the product of the two smallest of M, N, and L, and mnl is the product of all three.

REFERENCES

Bedinger, M.S., Pearson, F.J., Jr., Reed, J.E., Sniegocki, R.T., and Stone, C.G., 1979, The waters of Hot Springs National Park, Arkansas--Their nature and origin: U.S. Geological Survey Professional Paper 1044-C, 33 p.

Bennett, G.D., 1976, Introduction to ground-water hydraulics: U.S. Geological Survey Techniques of Water Resources Investigations, Book 3, Chapter B2, 172 p.

Faust, C. W., and Mercer, J. W., 1977, Finite difference model of two-dimensional single-, and two-phase heat transport in a porous medium--Version I: U.S. Geological Survey Open-File Report 77-234, 84 p.

Konikow, L.F., and Bredehoeft, J.D., 1978, Computer model of two-dimensional solute transport and dispersion in ground water: U.S. Geological Survey Techniques of Water Resources Investigations, Book 7, Chapter C2, 90 p.

Kreyszig, Erwin, 1972, Advanced engineering mathematics: New York, John Wiley and Sons, 888 p.

Larson, S.P., 1978, Direct solution algorithm for the two dimensional ground-water flow model: U.S. Geological Survey Open-File Report 79-202, 22 p.

Lohman, S.W., and others, 1972, Definitions of selected ground-water terms-revisions and conceptual refinements: U.S. Geological Survey Water-Supply Paper 1988, 21 p.

Sorey, M.L., 1978, Numerical modeling of liquid geothermal systems: U.S. Geological Survey Professional Paper 1044-D, 25 p.

Trescott, P.D., Pinder, G.F., and Larson, S.P., 1976, Finite-difference model for aquifer simulation in two dimensions, with results of numerical experiments: U.S. Geological Survey Techniques of Water Resources Investigations, Book 7, Chapter C1, 116 p.

Attachment 1. Program listing

Attachment 1.

```
      PROGRAM HOTWTR                                                                  HOTW  1
      LOGICAL LFIN,LSTOP,LHCPT,LVPT                                                   HOTW  2
      CHARACTER G8                                                                    HOTW  3
      CHARACTER*12 FILE,TEMP                                                          HOTW  4
      CHARACTER*18 FILE1,FILE2,FILE3,FILE4,FILE5,FILE6                                HOTW  5
      REAL KT                                                                         HOTW  6
      REAL*8 A                                                                        HOTW  7
      CHARACTER*8 XMESUR,ZMESUR                                                       HOTW  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,                           HOTW  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                              HOTW 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR                HOTW 11
      COMMON//A(253000)                                                               HOTW 12
      COMMON/G1/G1(32768)                                                             HOTW 13
      COMMON/G2/G2(32768)                                                             HOTW 14
      COMMON/G3/G3(32768)                                                             HOTW 15
      COMMON/G4/G4(32768)                                                             HOTW 16
      COMMON/G5/G5(32768)                                                             HOTW 17
      COMMON/G6/G6(32768)                                                             HOTW 18
      COMMON/G7/G7(32768)                                                             HOTW 19
      COMMON/G8/G8(131072)                                                            HOTW 20
      DATA LREC,LSTOP/4,.FALSE./                                                      HOTW 21
      LFIN=.FALSE.                                                                    HOTW 22
    1 WRITE(1,7)                                                                      HOTW 23
      READ(1,8)FILE                                                                   HOTW 24
    2 IBP=INDEX(FILE,' ')                                                             HOTW 25
      IF(IBP.EQ.0)THEN                                                                HOTW 26
           IBP=13                                                                     HOTW 27
      ELSE IF(IBP.EQ.1)THEN                                                           HOTW 28
           IF(FILE.EQ.'                    ')THEN                                     HOTW 29
                WRITE(1,9)                                                            HOTW 30
                GOTO1                                                                 HOTW 31
           ENDIF                                                                      HOTW 32
           TEMP=FILE                                                                  HOTW 33
           FILE=TEMP(2:12)                                                            HOTW 34
           GOTO2                                                                      HOTW 35
      ENDIF                                                                           HOTW 36
      FILE1=FILE(1:IBP-1)//'.INPUT'                                                   HOTW 37
      FILE2=FILE(1:IBP-1)//'.BAL'                                                     HOTW 38
      FILE3=FILE(1:IBP-1)//'.VELO'                                                    HOTW 39
      FILE4=FILE(1:IBP-1)//'.MOVE'                                                    HOTW 40
      FILE5=FILE(1:IBP-1)//'.PATH'                                                    HOTW 41
      FILE6=FILE(1:IBP-1)//'.PRINT'                                                   HOTW 42
      OPEN(5,FILE=FILE)                                                               HOTW 43
      READ(5,10) M,N,L,MAX,IREF,LHCPT,LVPT                                            HOTW 44
      MN=M*N                                                                          HOTW 45
      LMN=L*MN                                                                        HOTW 46
C     SET POINTERS FOR G1 ARRAY                                                       HOTW 47
      IHD=1                                                                           HOTW 48
      ITMP=IHD+LMN*2                                                                  HOTW 49
      IDX=ITMP+LMN*2                                                                  HOTW 50
      IDY=IDX+N                                                                       HOTW 51
      IDZ=IDY+M                                                                       HOTW 52
```

Attachment 1, continued

```
      IDISTX=IDZ+L                                                    HOTW 53
      IDISTZ=IDISTX+N                                                 HOTW 54
      IEND1=IDISTZ+L-1                                                HOTW 55
C     SET POINTERS FOR G2 ARRAY                                       HOTW 56
      IIH=1                                                           HOTW 57
      IIT=IIH+LMN                                                     HOTW 58
      INP=IIT+LMN                                                     HOTW 59
      IIR=INP+LMN                                                     HOTW 60
C     SET POINTERS FOR G4 ARRAY                                       HOTW 61
      IHC=1                                                           HOTW 62
      IP=IHC+LMN                                                      HOTW 63
      IPOR=IP+LMN                                                     HOTW 64
      IRCH=IPOR+LMN                                                   HOTW 65
      IRJ=IRCH+MN                                                     HOTW 66
      IDP=IRJ+N*3                                                     HOTW 67
      IHF=IDP+MN                                                      HOTW 68
      IEND4=IHF+MN-1                                                  HOTW 69
C     SET POINTERS FOR G7 ARRAY                                       HOTW 70
      IVX1=1                                                          HOTW 71
      IVX2=IVX1+LMN                                                   HOTW 72
      IVY1=IVX2+LMN                                                   HOTW 73
      IVY2=IVY1+LMN                                                   HOTW 74
      IVZ1=IVY2+LMN                                                   HOTW 75
      IVZ2=IVZ1+LMN                                                   HOTW 76
      IEND7=IVZ2+LMN-1                                                HOTW 77
C     SET POINTERS FOR G8 ARRAY                                       HOTW 78
      ISEC=1                                                          HOTW 79
      IF(M.GT.1)THEN                                                  HOTW 80
           ITIM=1                                                     HOTW 81
      ELSE                                                            HOTW 82
           ITIM=ISEC+LMN+MOD(LMN,4)                                   HOTW 83
      ENDIF                                                           HOTW 84
      IIDS=ITIM+4*LMN                                                 HOTW 85
      IPATH=IIDS+4*LMN                                                HOTW 86
      IF(IEND1.GT.32768)THEN                                          HOTW 87
           WRITE(1,11)IEND1                                           HOTW 88
           LSTOP=.TRUE.                                               HOTW 89
      END IF                                                          HOTW 90
      IF(IIR.GT.32768)THEN                                            HOTW 91
           WRITE(1,12)IIR                                             HOTW 92
           LSTOP=.TRUE.                                               HOTW 93
      END IF                                                          HOTW 94
      IF(IEND4.GT.32768)THEN                                          HOTW 95
           WRITE(1,13)IEND4                                           HOTW 96
           LSTOP=.TRUE.                                               HOTW 97
      END IF                                                          HOTW 98
      IF(IEND7.GT.32768)THEN                                          HOTW 99
           WRITE(1,14)IEND7                                           HOTW100
           LSTOP=.TRUE.                                               HOTW101
      END IF                                                          HOTW102
      IF(IPATH.GT.131072)THEN                                         HOTW103
           WRITE(1,15)IPATH                                           HOTW104
```

Attachment 1, continued

```
          LSTOP=.TRUE.                                                    HOTW105
        END IF                                                            HOTW106
        IF(LSTOP)GOTO6                                                    HOTW107
        WRITE(1,16)                                                       HOTW108
        OPEN(6,FILE=FILE1)                                                HOTW109
        CALL DATAIN(G4(IRCH),G1(ITMP),G4(IHF),G2(IIT),G4(IP),             HOTW110
      c G4(IPOR),G4(IHC),G4(IDP),G2(IIH),G1(IHD),G1(IDX),G1(IDY),G1(IDZ),HOTW111
      c G2(INP),G8(ISEC))                                                 HOTW112
      c                                                                   HOTW113
C       MORE POINTERS FOR G2 ARRAY                                        HOTW114
        IJR=IIR+NEQ                                                       HOTW115
        IKR=IJR+NEQ                                                       HOTW116
        IEND2=IKR+NEQ-1                                                   HOTW117
C       SET POINTERS FOR G3 ARRAY                                         HOTW118
        IQ=1                                                              HOTW119
        IEND3=NEQ*6                                                       HOTW120
C       SET POINTERS FOR G5 ARRAY                                         HOTW121
        IAU=1                                                             HOTW122
        IEND5=NU*12                                                       HOTW123
C       SET POINTERS FOR G6 ARRAY                                         HOTW124
        IDU=1                                                             HOTW125
        IRU=IDU+NU*2                                                      HOTW126
        IRL=IRU+NU*2                                                      HOTW127
        IEND6=IRL+NL*2-1                                                  HOTW128
        IENDA=NL*IBW                                                      HOTW129
        IF(IEND2.GT.32768)THEN                                            HOTW130
             WRITE(1,12)IEND2                                             HOTW131
             LSTOP=.TRUE.                                                 HOTW132
        END IF                                                            HOTW133
        IF(IEND3.GT.32768)THEN                                            HOTW134
             WRITE(1,17)IEND3                                             HOTW135
             LSTOP=.TRUE.                                                 HOTW136
        END IF                                                            HOTW137
        IF(IEND5.GT.32768)THEN                                            HOTW138
             WRITE(1,18)IEND5                                             HOTW139
             LSTOP=.TRUE.                                                 HOTW140
        END IF                                                            HOTW141
        IF(IEND6.GT.32768)THEN                                            HOTW142
             WRITE(1,19)IEND6                                             HOTW143
             LSTOP=.TRUE.                                                 HOTW144
        END IF                                                            HOTW145
        IF(IENDA.GT.253000)THEN                                           HOTW146
             WRITE(1,20)IENDA                                             HOTW147
             LSTOP=.TRUE.                                                 HOTW148
        END IF                                                            HOTW149
        IF(LSTOP)GOTO6                                                    HOTW150
        CALL PARTN(G2(INP),G2(IIR),G2(IJR),G2(IKR),G3(IQ))                HOTW151
        CLOSE(6)                                                          HOTW152
        OPEN(6,FILE=FILE2)                                                HOTW153
      3 WRITE(1,21)                                                       HOTW154
        CALL PA(G1(ITMP),G2(IIT),G2(IIH),G2(INP),G4(IP),G4(IHC),G3(IQ))   HOTW155
        IF(LFIN)GOTO4                                                     HOTW156
        WRITE(1,22)
```

Attachment 1, continued

```
      CALL WATEQ(G6(IRL),G6(IRU),G6(IDU),G5(IAU),                        HOTW157
     c G2(IIR),G2(IJR),G2(IKR),G2(INP),G4(IHC),G1(IDX),G1(IDY),G1(IDZ),  HOTW158
     c G2(IIH),G1(IHD),G4(IRCH))                                        HOTW159
      WRITE(1,23)                                                       HOTW160
      IGATE = IGATE + 1                                                 HOTW161
      WRITE(1,24) IGATE                                                 HOTW162
      CALL SOLVE(G6(IRL))                                               HOTW163
      CALL BACK(G2(IIR),G2(IJR),G2(IKR),G2(INP),G1(IHD),G6(IRL),        HOTW164
     c G6(IRU),G5(IAU),G6(IDU),DELV)                                    HOTW165
      DELH=DELV                                                         HOTW166
      WRITE(1,25)DELH                                                   HOTW167
      WRITE(6,26)IGATE,DELH                                             HOTW168
      IF(ABS(DELT).LT.TCONV.AND.ABS(DELH).LT.HCONV)LFIN=.TRUE.          HOTW169
      IF(IGATE.GE.MAX) LFIN=.TRUE.                                      HOTW170
    4 WRITE(1,27)                                                       HOTW171
      CALL COMPQ(G2(IIR),G2(IJR),G2(IKR),G2(IIH),G1(IHD),               HOTW172
     c G4(IHC),G1(IDX),G1(IDY),G1(IDZ),G3(IQ))                          HOTW173
      IF(LFIN)GOTO5                                                     HOTW174
      WRITE(1,28)                                                       HOTW175
      CALL HEATEQ(G6(IRL),G6(IRU),G6(IDU),G5(IAU),G2(IIR),G2(IJR),      HOTW176
     c G2(IKR),G2(INP), G1(ITMP),G2(IIT),G1(IDX),G1(IDY),G1(IDZ),       HOTW177
     c G4(IRCH),G4(IDP),G4(IHP),G3(IQ))                                 HOTW178
      WRITE(1,23)                                                       HOTW179
      IGATE = IGATE + 1                                                 HOTW180
      WRITE(1,24) IGATE                                                 HOTW181
      CALL SOLVE(G6(IRL))                                               HOTW182
      CALL BACK(G2(IIR),G2(IJR),G2(IKR),G2(INP),G1(ITMP),G6(IRL),       HOTW183
     c G6(IRU),G5(IAU),G6(IDU),DELV)                                    HOTW184
      DELT=DELV                                                         HOTW185
      WRITE(1,29)DELT                                                   HOTW186
      WRITE(6,30)IGATE,DELT                                             HOTW187
      IF(ABS(DELT).LT.TCONV.AND.ABS(DELH).LT.HCONV)LFIN=.TRUE.          HOTW188
      IF(IGATE.GE.MAX) LFIN=.TRUE.                                      HOTW189
      IF(LFIN)GOTO5                                                     HOTW190
      GOTO3                                                             HOTW191
    5 WRITE(1,31)                                                       HOTW192
      CALL BAL(G1(ITMP),G4(IHC),G1(IDX),G1(IDY),G1(IDZ),G2(IIT),G4(IHP),HOTW193
     c G4(IDP),G4(IRCH),G2(IIH),G1(IHD))                                HOTW194
      WRITE(1,32)                                                       HOTW195
      WRITE(6,32)                                                       HOTW196
      CLOSE(6)                                                          HOTW197
      IF(LVPT)OPEN(6,FILE=FILE3)                                        HOTW198
      CALL VELO(G2(IIH),G1(IHD),G4(IHC),G1(IDX),G1(IDY),                HOTW199
     c G1(IDZ),G7(IVX1),G7(IVX2),G7(IVY1),G7(IVY2),                     HOTW200
     c G7(IVZ1),G7(IVZ2),G4(IPOR),G4(IRCH))                             HOTW201
      IF(LVPT)CLOSE(6)                                                  HOTW202
      OPEN(6,FILE=FILE4)                                                HOTW203
      WRITE(1,33)                                                       HOTW204
      CALL MOVE(G2(IIH),G7(IVX1),G7(IVX2),G7(IVY1),                     HOTW205
     c G7(IVY2),G7(IVZ1),G7(IVZ2),G1(IDX),G1(IDY),G1(IDZ),              HOTW206
     c G8(ITIM),G8(IIDS))                                               HOTW207
      CLOSE(6)                                                          HOTW208
```

Attachment 1, continued

```
      IF(M.GT.1)GOTO6                                                   HOTW209
      NR=(131073-IPATH)/3                                               HOTW210
      NREC=NR/4                                                         HOTW211
      WRITE(1,34)                                                       HOTW212
      OPEN(6,FILE=FILE5)                                                HOTW213
      CALL PATH(G4(IRCH),G2(IIH),G7(IVX1),G7(IVX2),                     HOTW214
     c G7(IVZ1),G7(IVZ2),G1(IDX),G1(IDY),G1(IDZ),                       HOTW215
     c G1(IDISTX),G1(IDISTZ),G4(IRJ),NREC,G8(IPATH),                    HOTW216
     c G3(IQ),G2(INP))                                                  HOTW217
      CLOSE(6)                                                          HOTW218
      WRITE(1,35)                                                       HOTW219
      NR=NREC*4                                                         HOTW220
C     MORE POINTERS FOR G8 ARRAY                                        HOTW221
      IIPOS=IPATH+NR                                                    HOTW222
      ISRT=IIPOS+NR                                                     HOTW223
      IEND8=ISRT+NR-1                                                   HOTW224
      IF(IEND8.GT.131072)THEN                                           HOTW225
           WRITE(1,15)IEND8                                             HOTW226
           LSTOP=.TRUE.                                                 HOTW227
      END IF                                                            HOTW228
      IF(LSTOP)GOTO6                                                    HOTW229
      CALL SORT(G8(IIPOS),G8(IPATH),G8(ISRT),NREC,LREC)                 HOTW230
      WRITE(1,36)                                                       HOTW231
      OPEN(6,FILE=FILE6)                                                HOTW232
      CALL PRINT(G8(ITIM),G8(ISRT),G1(IHD),G2(IIH),G1(IDX),             HOTW233
     c G1(IDZ),G8(ISEC),NR)                                             HOTW234
      CLOSE(6)                                                          HOTW235
    6 STOP                                                              HOTW236
    7 FORMAT(' FILENAME=')                                              HOTW237
    8 FORMAT(A12)                                                       HOTW238
    9 FORMAT(' FILE NAME IS BLANK')                                     HOTW239
   10 FORMAT(5I3,2L1)                                                   HOTW240
   11 FORMAT(' SIZE OF G1 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW241
   12 FORMAT(' SIZE OF G2 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW242
   13 FORMAT(' SIZE OF G4 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW243
   14 FORMAT(' SIZE OF G7 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW244
   15 FORMAT(' SIZE OF G8 ARRAY = ',I6,' , DIMENSIONED AS 131072')      HOTW245
   16 FORMAT(' DATAIN')                                                 HOTW246
   17 FORMAT(' SIZE OF G3 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW247
   18 FORMAT(' SIZE OF G5 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW248
   19 FORMAT(' SIZE OF G6 ARRAY = ',I6,' , DIMENSIONED AS 32768')       HOTW249
   20 FORMAT(' SIZE OF A ARRAY = ',I6,' , DIMENSIONED AS 253000')       HOTW250
   21 FORMAT(' PA')                                                     HOTW251
   22 FORMAT(' WATEQ')                                                  HOTW252
   23 FORMAT(' SOLVE')                                                  HOTW253
   24 FORMAT(/1X,'ITERATION NUMBER ',I3/1X,20(1H-))                     HOTW254
   25 FORMAT(/1X,'MAX. CHANGE IN HEAD = ',1PE12.4)                      HOTW255
   26 FORMAT(/1X,'ITERATION NUMBER ',I3,                                HOTW256
     c ', MAX. CHANGE IN HEAD = ',1PE12.4)                              HOTW257
   27 FORMAT(' COMPQ')                                                  HOTW258
   28 FORMAT(' HEATEQ')                                                 HOTW259
   29 FORMAT(/1X,'MAX. CHANGE IN TEMP. = ',1PE12.4)                     HOTW260
```

Attachment 1, continued

```
   30 FORMAT(/1X,'ITERATION NUMBER ',I3,                    HOTW261
     c ', MAX. CHANGE IN TEMP. = ',1PE12.4)                 HOTW262
   31 FORMAT(' BAL')                                        HOTW263
   32 FORMAT(' VELO')                                       HOTW264
   33 FORMAT(' MOVE')                                       HOTW265
   34 FORMAT(' PATH')                                       HOTW266
   35 FORMAT(' SORT')                                       HOTW267
   36 FORMAT(' PRINT')                                      HOTW268
      END                                                   HOTW269
```

Attachment 1, continued

```
C     DATAIN - Reads input data.                                          DATA  1
      SUBROUTINE DATAIN(RCH,TMP,HF,IT,P,POR,HC,DP,IH,HD,DX,DY,DZ,         DATA  2
     c NP,SEC)                                                            DATA  3
      LOGICAL LFIN,LHCPT,LVPT                                             DATA  4
      REAL KT                                                             DATA  5
      REAL*8 HD,TMP                                                       DATA  6
      CHARACTER*8 XMESUR,ZMESUR                                           DATA  7
      DIMENSION RCH(M,N),TMP(M,N,L),HF(M,N),                              DATA  8
     cIT(M,N,L),P(M,N,L),POR(M,N,L),                                      DATA  9
     cHC(M,N,L),DP(M,N),                                                  DATA 10
     cIH(M,N,L),HD(M,N,L),NP(M,N,L)                                       DATA 11
      DIMENSION DX(N),DY(M),DZ(L)                                         DATA 12
      CHARACTER SEC(M,L)                                                  DATA 13
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,               DATA 14
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                  DATA 15
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR    DATA 16
      WRITE(6,8)M,N,L,MAX,IREF,LVPT,LHCPT                                 DATA 17
      READ(5,9) KT,SP,TTOP,TEMPR,TCONV,HCONV                              DATA 18
      WRITE(6,10) KT,SP,TTOP,TEMPR,TCONV,HCONV                            DATA 19
C     CONVERT FROM CONDUCTANCE UNITS TO BTU/(DAY*FT**2*(DEG.F/FT))        DATA 20
      KT=KT*5.8058                                                        DATA 21
      READ(5,9)DX                                                         DATA 22
      WRITE(6,11)                                                         DATA 23
      WRITE(6,12)DX                                                       DATA 24
      READ(5,9)DY                                                         DATA 25
      WRITE(6,13)                                                         DATA 26
      WRITE(6,12)DY                                                       DATA 27
      READ(5,9)DZ                                                         DATA 28
      WRITE(6,14)                                                         DATA 29
      WRITE(6,12)DZ                                                       DATA 30
      DO 1 K=1,L                                                          DATA 31
      DO 1 I=1,M                                                          DATA 32
    1 READ(5,15)(IT(I,J,K),J=1,N)                                         DATA 33
      WRITE(6,16)                                                         DATA 34
      DO 2 K=1,L                                                          DATA 35
      WRITE(6,17)K                                                        DATA 36
      DO 2 I=1,M                                                          DATA 37
    2 WRITE(6,18)(IT(I,J,K),J=1,N)                                        DATA 38
      DO 3 K=1,L                                                          DATA 39
      DO 3 I=1,M                                                          DATA 40
    3 READ(5,15)(IH(I,J,K),J=1,N)                                         DATA 41
      WRITE(6,19)                                                         DATA 42
      DO 4 K=1,L                                                          DATA 43
      WRITE(6,17)K                                                        DATA 44
      DO 4 I=1,M                                                          DATA 45
    4 WRITE(6,18)(IH(I,J,K),J=1,N)                                        DATA 46
      WRITE(6,20)                                                         DATA 47
      CALL READ(DP,M,N,1)                                                 DATA 48
      WRITE(6,21)                                                         DATA 49
      CALL READ(RCH,M,N,1)                                                DATA 50
      WRITE(6,22)                                                         DATA 51
      CALL READ(HF,M,N,1)
```

Attachment 1, continued

```
      WRITE(6,23)                                                        DATA 52
      CALL DREAD(HD,M,N,L)                                               DATA 53
      WRITE(6,24)                                                        DATA 54
      CALL DREAD(TMP,M,N,L)                                              DATA 55
      WRITE(6,25)                                                        DATA 56
      CALL READ(P,M,N,L)                                                 DATA 57
      WRITE(6,26)                                                        DATA 58
      CALL READ(POR,M,N,L)                                               DATA 59
      IF(M.EQ.1)THEN                                                     DATA 60
           WRITE(6,27)                                                   DATA 61
           DO 5 K=1,L                                                    DATA 62
           READ(5,28)(SEC(J,K),J=1,N)                                    DATA 63
           WRITE(6,29)(SEC(J,K),J=1,N)                                   DATA 64
    5      CONTINUE                                                      DATA 65
           READ(5,30)XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR           DATA 66
           WRITE(6,31)XSCALE,DINCHX,XMESUR,ZSCALE,DINCHZ,ZMESUR          DATA 67
      ENDIF                                                              DATA 68
      CLOSE(5)                                                           DATA 69
      IGATE = 0                                                          DATA 70
      DO 6 J=1,N                                                         DATA 71
      DXJ=DX(J)                                                          DATA 72
      DO 6 I=1,M                                                         DATA 73
      XMUL1=DXJ*DY(I)                                                    DATA 74
      XMUL2=XMUL1/(12.*365.25)                                           DATA 75
      XMUL3=XMUL1*.31853                                                 DATA 76
C     CONVERT FROM HFU TO BTU/DAY                                        DATA 77
      HF(I,J)=HF(I,J)*XMUL3                                              DATA 78
C     CONVERT FROM IN/YR TO FT**3/DAY                                    DATA 79
      RCH(I,J)=RCH(I,J)*XMUL2                                            DATA 80
C     CONVERT FROM FEET TO (BTU/DAY)/DEG.F                               DATA 81
      IF(DP(I,J).GT.0.)DP(I,J)=KT*XMUL1/DP(I,J)                          DATA 82
      DO 6 K=1,L                                                         DATA 83
      HC(I,J,K) = 0.                                                     DATA 84
    6 CONTINUE                                                           DATA 85
      DELH=HCONV                                                         DATA 86
      DELT=TCONV                                                         DATA 87
      DO 7 K=1,L                                                         DATA 88
      DO 7 J=1,N                                                         DATA 89
      DO 7 I=1,M                                                         DATA 90
      NP(I,J,K)=0                                                        DATA 91
    7 CONTINUE                                                           DATA 92
      CALL NUMBER(IT,NP)                                                 DATA 93
      RETURN                                                             DATA 94
    8 FORMAT(' NUMBER OF ROWS = ',I3/' NUMBER OF COLS = ',I3             DATA 95
     c /' NUMBER OF LAYERS = ',I3/' MAXIMUM ITERATIONS = ',I3            DATA 96
     c /' REFERENCE TEMP. FOR HYD. COND. = ',I3,' DEG. C'                DATA 97
     c /' VELOCITY PRINT OPTION = ',L1                                   DATA 98
     c /' HYD. COND. PRINT OPTION = ',L1)                                DATA 99
    9 FORMAT(16F5.0)                                                     DATA100
   10 FORMAT(/' THERMAL CONDUCTIVITY=',1PE12.4,                          DATA101
     c ' C.U. - 10**-3 CAL./(CM.*S.',                                    DATA102
     c '*DEG. C)'/' SPECIFIC HEAT=',E12.4,' BTU/CUBIC FT/DEG. F'/        DATA103
```

Attachment 1, continued

```
     c ' SURFACE TEMP.=',E12.4,' DEG. F'/                                DATA104
     c ' RECHARGE TEMP. = ',E12.4,' DEG. F'/                             DATA105
     c ' TEMP. CONV. CRITERION = ',E12.4,' DEG. F'/                      DATA106
     c ' HEAD CONV. CRITERION = ',E12.4,' FEET')                         DATA107
    11 FORMAT(/' HORIZONTAL (X) NODE SPACING, FEET'/1H ,33(1H-))         DATA108
    12 FORMAT(1X,1P10E12.4)                                              DATA109
    13 FORMAT(/' HORIZONTAL (Y) NODE SPACING, FEET'/1H ,33(1H-))         DATA110
    14 FORMAT(/' VERTICAL NODE SPACING, FEET'/1H ,27(1H-))               DATA111
    15 FORMAT(80I1)                                                      DATA112
    16 FORMAT(/' NODE LEVEL FOR TEMP.'/1H ,20(1H-))                      DATA113
    17 FORMAT(/' LAYER ',I2/1H ,8(1H-))                                  DATA114
    18 FORMAT(1H ,80I1)                                                  DATA115
    19 FORMAT(/' NODE LEVEL FOR HEAD'/1H ,19(1H-))                       DATA116
    20 FORMAT(/' DEPTH BELOW LAND SURFACE, FEET'/1H ,30(1H-))            DATA117
    21 FORMAT(/' RECHARGE RATE, IN/YR'/1H ,20(1H-))                      DATA118
    22 FORMAT(/' HEAT FLOW, H.F.U. - 10**-6 CAL./(SQ. CM.*S.)'/1H ,      DATA119
      c 44(1H-))                                                         DATA120
    23 FORMAT(/' INITIAL HEAD, FEET'/1H ,18(1H-))                        DATA121
    24 FORMAT(/' INITIAL TEMPERATURE, DEG. F'/1H ,27(1H-))               DATA122
    25 FORMAT(/' HYDRAULIC CONDUCTIVITY (FT./DAY) AT REF. TEMP.'/        DATA123
      c 1H ,46(1H-))                                                     DATA124
    26 FORMAT(/' POROSITY'/1H ,8(1H-))                                   DATA125
    27 FORMAT(/' CROSS SECTION'/1H ,13(1H-))                             DATA126
    28 FORMAT(80A1)                                                      DATA127
    29 FORMAT(1X,80A1)                                                   DATA128
    30 FORMAT(4F10.0,A8,2X,A8)                                           DATA129
    31 FORMAT(/' HORIZONTAL SCALE UNIT = ',G12.5,' FEET'                 DATA130
     c /' NUMBER PER INCH= ',G12.5/' NAME = ',A8                         DATA131
     c /' VERTICAL SCALE UNIT = ',G12.5,' FEET'                          DATA132
     c /' NUMBER PER INCH= ',G12.5/' NAME = ',A8)                        DATA133
       END                                                               DATA134
```

Attachment 1, continued

```
C     READ - Reads REAL*4 arrays.                                    READ  1
      SUBROUTINE READ(ARRAY,M,N,KL)                                  READ  2
      DIMENSION ARRAY(1),TEMP(100)                                   READ  3
      MN=M*N                                                         READ  4
      READ(5,4)CONS,IVAR                                             READ  5
      IF(IVAR.EQ.0)THEN                                              READ  6
           NARY=MN*KL                                                READ  7
           DO 1 IARY=1,NARY                                          READ  8
    1      ARRAY(IARY)=CONS                                          READ  9
           WRITE(6,5)CONS                                            READ 10
      ELSE                                                           READ 11
           KK=-M-MN                                                  READ 12
           DO 3 K=1,KL                                               READ 13
           KK=KK+MN                                                  READ 14
           II=KK                                                     READ 15
           IF(KL.GT.1)WRITE(6,6)K                                    READ 16
           DO 3 I=1,M                                                READ 17
           II=II+1                                                   READ 18
           IJK=II                                                    READ 19
           READ(5,7)(TEMP(J),J=1,N)                                  READ 20
           DO 2 J=1,N                                                READ 21
           IJK=IJK+M                                                 READ 22
C     IJK=I+(J-1)*M+(K-1)*M*N                                        READ 23
           ARRAY(IJK)=TEMP(J)*CONS                                   READ 24
    2      TEMP(J)=ARRAY(IJK)                                        READ 25
           WRITE(6,8)(TEMP(J),J=1,N)                                 READ 26
    3      CONTINUE                                                  READ 27
      ENDIF                                                          READ 28
      RETURN                                                         READ 29
    4 FORMAT(F10.0,I5)                                               READ 30
    5 FORMAT(' UNIFORMLY =',1PE12.3)                                 READ 31
    6 FORMAT(/' LAYER ',I2/1H ,8(1H-))                               READ 32
    7 FORMAT(20F4.0)                                                 READ 33
    8 FORMAT(1H ,1P10E12.3)                                          READ 34
      END
```

Attachment 1, continued

```
C     DREAD - Reads REAL*8 arrays.                                    DREA  1
      SUBROUTINE DREAD(ARRAY,M,N,KL)                                  DREA  2
      REAL*8 ARRAY(1),TEMP(100)                                       DREA  3
      MN=M*N                                                          DREA  4
      READ(5,4)CONS,IVAR                                              DREA  5
      IF(IVAR.EQ.0)THEN                                               DREA  6
           NARY=MN*KL                                                 DREA  7
           DO 1 IARY=1,NARY                                           DREA  8
    1      ARRAY(IARY)=CONS                                           DREA  9
           WRITE(6,5)CONS                                             DREA 10
      ELSE                                                            DREA 11
           KK=-M-MN                                                   DREA 12
           DO 3 K=1,KL                                                DREA 13
           KK=KK+MN                                                   DREA 14
           II=KK                                                      DREA 15
           IF(KL.GT.1)WRITE(6,6)K                                     DREA 16
           DO 3 I=1,M                                                 DREA 17
           II=II+1                                                    DREA 18
           IJK=II                                                     DREA 19
           READ(5,7)(TEMP(J),J=1,N)                                   DREA 20
           DO 2 J=1,N                                                 DREA 21
           IJK=IJK+M                                                  DREA 22
C     IJK=I+(J-1)*M+(K-1)*M*N                                         DREA 23
           ARRAY(IJK)=TEMP(J)*CONS                                    DREA 24
    2      TEMP(J)=ARRAY(IJK)                                         DREA 25
           WRITE(6,8)(TEMP(J),J=1,N)                                  DREA 26
    3      CONTINUE                                                   DREA 27
      ENDIF                                                           DREA 28
      RETURN                                                          DREA 29
    4 FORMAT(F10.0,I5)                                                DREA 30
    5 FORMAT(' UNIFORMLY =',1PE12.3)                                  DREA 31
    6 FORMAT(/' LAYER ',I2/1H ,8(1H-))                                DREA 32
    7 FORMAT(20F4.0)                                                  DREA 33
    8 FORMAT(1H ,1P10E12.3)                                           DREA 34
      END
```

Attachment 1, continued

```
C     NUMBER - Numbers the equations, sets pointers from model grid
C        to equations.                                                    NUMB  1
      SUBROUTINE NUMBER(IT,NP)                                           NUMB  2
      LOGICAL LFIN,LHCPT,LVPT                                            NUMB  3
      REAL KT                                                            NUMB  4
      CHARACTER*8 XMESUR,ZMESUR                                          NUMB  5
      DIMENSION IT(1),NP(1)                                              NUMB  6
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,              NUMB  7
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                 NUMB  8
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR    NUMB  9
      I1L=MAX0(M,N,L)                                                    NUMB 10
      I3L=MIN0(M,N,L)                                                    NUMB 11
      IF(M.EQ.I1L)THEN                                                   NUMB 12
           I1I=1                                                         NUMB 13
           IF(L.EQ.I3L)THEN                                              NUMB 14
                I2L=N                                                    NUMB 15
                I2I=M                                                    NUMB 16
                I3I=MN                                                   NUMB 17
           ELSE                                                          NUMB 18
                I2L=L                                                    NUMB 19
                I2I=MN                                                   NUMB 20
                I3I=M                                                    NUMB 21
           ENDIF                                                         NUMB 22
      ELSE IF(N.EQ.I1L)THEN                                              NUMB 23
           I1I=M                                                         NUMB 24
           IF(L.EQ.I3L)THEN                                              NUMB 25
                I2L=M                                                    NUMB 26
                I2I=1                                                    NUMB 27
                I3I=MN                                                   NUMB 28
           ELSE                                                          NUMB 29
                I2L=L                                                    NUMB 30
                I2I=MN                                                   NUMB 31
                I3I=1                                                    NUMB 32
           ENDIF                                                         NUMB 33
      ELSE                                                               NUMB 34
           I1I=MN                                                        NUMB 35
           IF(N.EQ.I3L)THEN                                              NUMB 36
                I2L=M                                                    NUMB 37
                I2I=1                                                    NUMB 38
                I3I=M                                                    NUMB 39
           ELSE                                                          NUMB 40
                I2L=N                                                    NUMB 41
                I2I=M                                                    NUMB 42
                I3I=1                                                    NUMB 43
           ENDIF                                                         NUMB 44
      ENDIF                                                              NUMB 45
      IBW=2*I2L*I3L+1                                                    NUMB 46
      MDM=IBW/2                                                          NUMB 47
      IEQ=0                                                              NUMB 48
      DO 2 NC=1,2                                                        NUMB 49
      IF(NC.EQ.1)THEN                                                    NUMB 50
           I3C=2
```

Attachment 1, continued

```
          I3INC=2*I3I                                                    NUMB 51
                                                                         NUMB 52
        ELSE                                                             NUMB 53
          I3S=1                                                          NUMB 54
          I3C=1                                                          NUMB 55
          I3INC=I3I                                                      NUMB 56
        ENDIF                                                            NUMB 57
        II1=-M-M*N                                                       NUMB 58
        DO 1 I1=1,I1L                                                    NUMB 59
        II1=II1+I1I                                                      NUMB 60
        II2=II1                                                          NUMB 61
        DO 1 I2=1,I2L                                                    NUMB 62
        II2=II2+I2I                                                      NUMB 63
        IF(NC.EQ.1)THEN                                                  NUMB 64
          IF(MOD(I1+I2,2).EQ.0)THEN                                      NUMB 65
            I3S=1                                                        NUMB 66
            II3=II2-I3I                                                  NUMB 67
          ELSE                                                           NUMB 68
            I3S=2                                                        NUMB 69
            II3=II2                                                      NUMB 70
          ENDIF                                                          NUMB 71
        ELSE                                                             NUMB 72
          II3=II2                                                        NUMB 73
        ENDIF                                                            NUMB 74
        DO 1 I3=I3S,I3L,I3C                                              NUMB 75
        II3=II3+I3INC                                                    NUMB 76
C       II3=I1*I1I+I2*I2I+I3*I3I-M-M*N                                   NUMB 77
        IF(IT(II3).NE.1)GOTO1                                            NUMB 78
        IF(NP(II3).NE.0)GOTO1                                            NUMB 79
        IEQ=IEQ+1                                                        NUMB 80
        NP(II3)=IEQ                                                      NUMB 81
      1 CONTINUE                                                         NUMB 82
        IF(NC.EQ.1)THEN                                                  NUMB 83
          NU=IEQ                                                         NUMB 84
        ELSE                                                             NUMB 85
          NEQ=IEQ                                                        NUMB 86
          NL=NEQ-NU                                                      NUMB 87
        ENDIF                                                            NUMB 88
      2 CONTINUE                                                         NUMB 89
        RETURN                                                           NUMB 90
        END
```

Attachment 1, continued

```
C     PARTN - Sets pointers from equation to model grid, zeros Q array.
      SUBROUTINE PARTN(NP,IR,JR,KR,Q)
      LOGICAL LFIN,LHCPT,LVPT
      CHARACTER*8 XMESUR,ZMESUR
      DIMENSION NP(M,N,L),IR(NEQ),JR(NEQ),KR(NEQ),Q(NEQ,6)
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR
      DO 1 K=1,L
      DO 1 J=1,N
      DO 1 I=1,M
      IEQ=NP(I,J,K)
      IF(IEQ.EQ.0)GOTO1
      IR(IEQ)=I
      JR(IEQ)=J
      KR(IEQ)=K
    1 CONTINUE
      WRITE(6,3)
      WRITE(6,4)(IEQ,IR(IEQ),JR(IEQ),IEQR(IEQ),IEQ=1,NEQ)
      DO 2 JJ=1,6
      DO 2 IEQ=1,NEQ
    2 Q(IEQ,JJ)=0.
      RETURN
    3 FORMAT(/' BLOCK NUMBER AND ROW, COL., AND LAYER NUMBER'/
     c 1H ,44(1H-))
    4 FORMAT(1H ,9(I4,'=',I2,',',I2,',',I2,1H ))
      END
```

PART 1 PART 2 PART 3 PART 4 PART 5 PART 6 PART 7 PART 8 PART 9 PART 10 PART 11 PART 12 PART 13 PART 14 PART 15 PART 16 PART 17 PART 18 PART 19 PART 20 PART 21 PART 22 PART 23 PART 24 PART 25 PART 26

Attachment 1, continued

```
C     PA - Adjusts hydraulic conductivity for temperature change.
      SUBROUTINE PA(TMP,IT,IH,NP,P,HC,Q)                                PA  1
      LOGICAL LFIN,LHCPT,LVPT                                           PA  2
      REAL KT                                                           PA  3
      REAL*8 TMP                                                        PA  4
      CHARACTER*8 XMESUR,ZMESUR                                         PA  5
      DIMENSION TMP(M,N,L),IT(M,N,L),IH(M,N,L),NP(M,N,L),               PA  6
     c P(M,N,L),HC(M,N,L)                                               PA  7
      DIMENSION IC(6),JC(6),KC(6),IV(6),IVT(6)                          PA  8
      DIMENSION Q(NEQ,6),XKV(100)                                       PA  9
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,             PA 10
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                PA 11
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR  PA 12
      DATA XKV/1.7322,1.6741,1.6194,1.5676,1.5189,1.4727,1.4289,        PA 13
     c 1.3874,1.3479,1.3101,1.2740,1.2396,1.2069,1.1757,1.1457,1.1168,  PA 14
     c 1.0889,1.0618,1.0357,1.0105,.9863,.9629,.9403,.9186,.8976,       PA 15
     c .8774,.8581,.8395,.8214,.8039,.7871,.7668,.7551,.7399,.7251,     PA 16
     c .7109,.6971,.6839,.6711,.6587,.6468,.6352,.6240,.6132,.6029,     PA 17
     c .5929,.5832,.5739,.5647,.5558,.5473,.5389,.5307,.5225,.5146,     PA 18
     c .5069,.4994,.4921,.4849,.4779,.4711,.4644,.4579,.4516,.4455,     PA 19
     c .4394,.4334,.4276,.4219,.4164,.4110,.4057,.4005,.3954,.3904,     PA 20
     c .3855,.3808,.3762,.3718,.3674,.3631,.3589,.3547,.3504,.3466,     PA 21
     c .3427,.3389,.3352,.3316,.3280,.3245,.3211,.3178,.3145,.3113,     PA 22
     c .3082,.3051,.3020,.2991,.2962/                                   PA 23
      DATA IC/+1,-1,0,0,0,0/,JC/0,0,+1,-1,0,0/                          PA 24
      DATA KC/0,0,0,0,+1,-1/,IVT/6*1/                                   PA 25
      IVT(1)=M                                                          PA 26
      IVT(3)=N                                                          PA 27
      IVT(5)=L                                                          PA 28
      DO 2 K=1,L                                                        PA 29
      DO 2 J=1,N                                                        PA 30
      DO 2 I=1,M                                                        PA 31
      IF(IH(I,J,K).NE.3)THEN                                            PA 32
           TC=(TMP(I,J,K)-32.)/1.8                                      PA 33
           IF((IH(I,J,K).EQ.2).AND.(IT(I,J,K).EQ.3))THEN                PA 34
                TSUM=0.                                                 PA 35
                QSUM=0.                                                 PA 36
                IV(1)=I                                                 PA 37
                IV(2)=I                                                 PA 38
                IV(3)=J                                                 PA 39
                IV(4)=J                                                 PA 40
                IV(5)=K                                                 PA 41
                IV(6)=K                                                 PA 42
                DO 1 II=1,6                                             PA 43
                IF(IV(II).NE.IVT(II))THEN                               PA 44
                     IIC=I+IC(II)                                       PA 45
                     JJC=J+JC(II)                                       PA 46
                     KKC=K+KC(II)                                       PA 47
                     NPIJK=NP(IIC,JJC,KKC)                              PA 48
                     IF(NPIJK.NE.0)THEN                                 PA 49
                          QQ=Q(NPIJK,II)                                PA 50
                          IF(QQ.LT.0.)THEN                              PA 51
```

Attachment 1, continued

```
                                TSUM=TSUM-QQ*TMP(IIC,JJC,KKC)                    PA 52
                                QSUM=QSUM-QQ                                     PA 53
                            ENDIF                                                PA 54
                        ENDIF                                                    PA 55
                    ENDIF                                                        PA 56
                ENDIF                                                            PA 57
    1           CONTINUE                                                         PA 58
                IF(QSUM.GT.0.)THEN                                               PA 59
                    TAVE=TSUM/QSUM                                               PA 60
                    TC=(TAVE-32.)/1.8                                            PA 61
                ENDIF                                                            PA 62
            ENDIF                                                                PA 63
            IF(TC.GT.100.)THEN                                                   PA 64
                TC=100.                                                          PA 65
            ELSE IF(TC.LT.1.)THEN                                                PA 66
                TC=1.                                                            PA 67
            ENDIF                                                                PA 68
            IXKV=TC                                                              PA 69
            IXKV1=IXKV+1                                                         PA 70
            IF(IXKV1.GT.100)IXKV1=100                                            PA 71
            XKVTC=XKV(IXKV)+(TC-IXKV)*(XKV(IXKV1)-XKV(IXKV))                     PA 72
            HC(I,J,K)=XKV(IREF)/XKVTC*P(I,J,K)                                   PA 73
        ENDIF                                                                    PA 74
    2   CONTINUE                                                                 PA 75
        RETURN                                                                   PA 76
        END
```

Attachment 1, continued

```
C     WATEQ - Computes coefficient matrix and known vector for
C           ground-water-flow equations.
      SUBROUTINE WATEQ(RL,RU,DU,AU,IR,JR,KR,NP,HC,DX,DY,DZ,                WATE  1
     c IH,HD,RCH)                                                          WATE  2
      LOGICAL LFIN,LHCPT,LVPT                                              WATE  3
      REAL KT                                                              WATE  4
      REAL*8 HD                                                            WATE  5
      REAL*8 A,RL,RU,DU,AU,AL                                              WATE  6
      REAL*8 REQ,DCOEF,ACOEF,GCOEF,BCOEF,FCOEF,CCOEF,ECOEF                 WATE  7
      CHARACTER*8 XMESUR,ZMESUR                                            WATE  8
      DIMENSION RL(NL),RU(NU),DU(NU),AU(6,NU),AL(6),                       WATE  9
     c IR(NEQ),JR(NEQ),KR(NEQ),NP(M,N,L),HC(M,N,L),DX(N),DY(M),DZ(L),      WATE 10
     c IH(M,N,L),HD(M,N,L),RCH(M,N)                                        WATE 11
      DIMENSION IC(6),JC(6),KC(6)                                          WATE 12
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,                WATE 13
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                   WATE 14
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR     WATE 15
      COMMON//A(253000)                                                    WATE 16
      DATA IC/-1,+1,0,0,0,0/,JC/0,0,-1,+1,0,0/,KC/0,0,0,0,-1,+1/           WATE 17
      DO 1 IEQ=1,NU                                                        WATE 18
      DU(IEQ)=0.                                                           WATE 19
      RU(IEQ)=0.                                                           WATE 20
      DO 1 JJ=1,6                                                          WATE 21
    1 AU(JJ,IEQ)=0.                                                        WATE 22
      DO 2 JJ=1,6                                                          WATE 23
    2 AL(JJ)=0.                                                            WATE 24
      DO 3 IL=1,NL                                                         WATE 25
    3 RL(IL)=0.                                                            WATE 26
      IL=0                                                                 WATE 27
      IA=-IBW                                                              WATE 28
      MD=MDM+1                                                             WATE 29
      DO 7 IEQ=1,NEQ                                                       WATE 30
      IF(IEQ.GT.NU)IL=IL+1                                                 WATE 31
      I=IR(IEQ)                                                            WATE 32
      J=JR(IEQ)                                                            WATE 33
      K=KR(IEQ)                                                            WATE 34
      HCIJK=HC(I,J,K)                                                      WATE 35
      DYI=DY(I)                                                            WATE 36
      DXJ=DX(J)                                                            WATE 37
      DZK=DZ(K)                                                            WATE 38
      AREAY=2.*DXJ*DZK                                                     WATE 39
      AREAX=2.*DYI*DZK                                                     WATE 40
      AREAZ=2.*DXJ*DYI                                                     WATE 41
      DCOEF=0.                                                             WATE 42
      REQ=0.                                                               WATE 43
      IF(I.NE.1)THEN                                                       WATE 44
           IHIM1=IH(I-1,J,K)                                               WATE 45
           IF(IHIM1.NE.3)THEN                                              WATE 46
                ACOEF=HC(I-1,J,K)*HCIJK/(HC(I-1,J,K)*DYI+HCIJK*DY(I-1))    WATE 47
     c          *AREAY                                                     WATE 48
                DCOEF=DCOEF-ACOEF                                          WATE 49
                IF(IHIM1.NE.2)THEN                                         WATE 50
```

Attachment 1, continued

```
                    IF(IEQ.GT.NU)THEN                                                    WATE 51
                        AL(1)=ACOEF                                                      WATE 52
                    ELSE                                                                 WATE 53
                        AU(1,IEQ)=ACOEF                                                  WATE 54
                    ENDIF                                                                WATE 55
                ENDIF                                                                    WATE 56
                REQ=REQ-ACOEF*HD(I-1,J,K)                                                WATE 57
            ENDIF                                                                        WATE 58
        ENDIF                                                                            WATE 59
        IF(J.NE.1)THEN                                                                   WATE 60
            IHJM1=IH(I,J-1,K)                                                            WATE 61
            IF(IHJM1.NE.3)THEN                                                           WATE 62
                BCOEF=HC(I,J-1,K)*HCIJK/(HC(I,J-1,K)*DXJ+HCIJK*DX(J-1))                  WATE 63
c               *AREAX                                                                   WATE 64
                DCOEF=DCOEF-BCOEF                                                        WATE 65
                IF(IHJM1.NE.2)THEN                                                       WATE 66
                    IF(IEQ.GT.NU)THEN                                                    WATE 67
                        AL(3)=BCOEF                                                      WATE 68
                    ELSE                                                                 WATE 69
                        AU(3,IEQ)=BCOEF                                                  WATE 70
                    ENDIF                                                                WATE 71
                ENDIF                                                                    WATE 72
                REQ=REQ-BCOEF*HD(I,J-1,K)                                                WATE 73
            ENDIF                                                                        WATE 74
        ENDIF                                                                            WATE 75
        IF(J.NE.M)THEN                                                                   WATE 76
            IHJP1=IH(I,J+1,K)                                                            WATE 77
            IF(IHJP1.NE.3)THEN                                                           WATE 78
                FCOEF=HC(I,J+1,K)*HCIJK/(HC(I,J+1,K)*DXJ+HCIJK*DX(J+1))                  WATE 79
c               *AREAX                                                                   WATE 80
                DCOEF=DCOEF-FCOEF                                                        WATE 81
                IF(IHJP1.NE.2)THEN                                                       WATE 82
                    IF(IEQ.GT.NU)THEN                                                    WATE 83
                        AL(4)=FCOEF                                                      WATE 84
                    ELSE                                                                 WATE 85
                        AU(4,IEQ)=FCOEF                                                  WATE 86
                    ENDIF                                                                WATE 87
                ENDIF                                                                    WATE 88
                REQ=REQ-FCOEF*HD(I,J+1,K)                                                WATE 89
            ENDIF                                                                        WATE 90
        ENDIF                                                                            WATE 91
        IF(I.NE.M)THEN                                                                   WATE 92
            IHIP1=IH(I+1,J,K)                                                            WATE 93
            IF(IHIP1.NE.3)THEN                                                           WATE 94
                GCOEF=HC(I+1,J,K)*HCIJK/(HC(I+1,J,K)*DYI+HCIJK*DY(I+1))                  WATE 95
c               *AREAY                                                                   WATE 96
                DCOEF=DCOEF-GCOEF                                                        WATE 97
                IF(IHIP1.NE.2)THEN                                                       WATE 98
                    IF(IEQ.GT.NU)THEN                                                    WATE 99
                        AL(2)=GCOEF                                                      WATE100
                    ELSE                                                                 WATE101
                        AU(2,IEQ)=GCOEF                                                  WATE102
```

Attachment 1, continued

```
                ENDIF                                                    WATE103
              ENDIF                                                      WATE104
              REQ=REQ-GCOEF*HD(I+1,J,K)                                  WATE105
            ENDIF                                                        WATE106
          ENDIF                                                          WATE107
          IF(K.NE.1)THEN                                                 WATE108
            IHKM1=IH(I,J,K-1)                                            WATE109
            IF(IHKM1.NE.3)THEN                                           WATE110
              CCOEF=HC(I,J,K-1)*HCIJK/(HC(I,J,K-1)*DZK+HCIJK*DZ(K-1))    WATE111
     c        *AREAZ                                                      WATE112
              DCOEF=DCOEF-CCOEF                                          WATE113
              IF(IHKM1.NE.2)THEN                                         WATE114
                IF(IEQ.GT.NU)THEN                                        WATE115
                  AL(5)=CCOEF                                            WATE116
                ELSE                                                     WATE117
                  AU(5,IEQ)=CCOEF                                        WATE118
                ENDIF                                                    WATE119
              ELSE                                                       WATE120
                REQ=REQ-RCH(I,J)                                         WATE121
              ENDIF                                                      WATE122
              REQ=REQ-CCOEF*HD(I,J,K-1)                                  WATE123
            ELSE                                                         WATE124
              REQ=REQ-RCH(I,J)                                           WATE125
            ENDIF                                                        WATE126
          ELSE                                                           WATE127
            REQ=REQ-RCH(I,J)                                             WATE128
          ENDIF                                                          WATE129
          IF(K.NE.L)THEN                                                 WATE130
            IHKP1=IH(I,J,K+1)                                            WATE131
            IF(IHKP1.NE.3)THEN                                           WATE132
              ECOEF=HC(I,J,K+1)*HCIJK/(HC(I,J,K+1)*DZK+HCIJK*DZ(K+1))    WATE133
     c        *AREAZ                                                      WATE134
              DCOEF=DCOEF-ECOEF                                          WATE135
              IF(IHKP1.NE.2)THEN                                         WATE136
                IF(IEQ.GT.NU)THEN                                        WATE137
                  AL(6)=ECOEF                                            WATE138
                ELSE                                                     WATE139
                  AU(6,IEQ)=ECOEF                                        WATE140
                ENDIF                                                    WATE141
              ENDIF                                                      WATE142
              REQ=REQ-ECOEF*HD(I,J,K+1)                                  WATE143
            ENDIF                                                        WATE144
          ENDIF                                                          WATE145
          IF(IEQ.LE.NU)THEN                                              WATE146
            DU(IEQ)=DCOEF                                                WATE147
            RU(IEQ)=REQ-DCOEF*HD(I,J,K)                                  WATE148
          ELSE                                                           WATE149
            RL(IL)=REQ-DCOEF*HD(I,J,K)                                   WATE150
            IA=IA+IBW                                                    WATE151
            DO 4 ISUB=IA+1,IA+IBW                                        WATE152
     4      A(ISUB)=0.                                                   WATE153
            IMD=IA+MD                                                    WATE154
```

Attachment 1, continued

```
            A(IMD)=DCOEF                                                    WATE155
            DO 6 II=1,6                                                     WATE156
            IIC=I+IC(II)                                                    WATE157
            IF(IIC.LE.0.OR.IIC.GT.M)GOTO6                                   WATE158
            JJC=J+JC(II)                                                    WATE159
            IF(JJC.LE.0.OR.JJC.GT.N)GOTO6                                   WATE160
            KKC=K+KC(II)                                                    WATE161
            IF(KKC.LE.0.OR.KKC.GT.L)GOTO6                                   WATE162
            NPQ=NP(IIC,JJC,KKC)                                             WATE163
            IF(NPQ.EQ.0)GOTO6                                               WATE164
            DO 5 JJ=1,6                                                     WATE165
            ICC=IIC+IC(JJ)                                                  WATE166
            IF(ICC.LE.0.OR.ICC.GT.M)GOTO5                                   WATE167
            JCC=JJC+JC(JJ)                                                  WATE168
            IF(JCC.LE.0.OR.JCC.GT.N)GOTO5                                   WATE169
            KCC=KKC+KC(JJ)                                                  WATE170
            IF(KCC.LE.0.OR.KCC.GT.L)GOTO5                                   WATE171
            NPP=NP(ICC,JCC,KCC)                                             WATE172
            IF(NPP.EQ.0)GOTO5                                               WATE173
            NN=IA+MD-IEQ+NPP                                                WATE174
            A(NN)=A(NN)-AU(JJ,NPQ)*AL(II)/DU(NPQ)                           WATE175
                                                                            WATE176
5           CONTINUE                                                        WATE177
            RL(IL)=RL(IL)-RU(NPQ)*AL(II)/DU(NPQ)                            WATE178
6           CONTINUE                                                        WATE179
          ENDIF                                                             WATE180
7         CONTINUE                                                          WATE181
          RETURN                                                            WATE182
          END
```

Attachment 1, continued

```
C     SOLVE - Reduces coefficient matrix to upper-diagonal form
C        and back substitutes to compute values for the unknowns.
      SUBROUTINE SOLVE(RL)                                              SOLV  1
C     ASYMMETRIC BAND MATRIX EQUATION SOLVER                            SOLV  2
C     ORIGINALLY PROGRAMED BY JAMES O. DUGUID                           SOLV  3
      LOGICAL LFIN,LHCPT,LVPT                                           SOLV  4
      REAL KT                                                           SOLV  5
      REAL*8 A,RL,PIVOT,C,SUM                                           SOLV  6
      CHARACTER*8 XMESUR,ZMESUR                                         SOLV  7
      DIMENSION  RL(NL)                                                 SOLV  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,             SOLV  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                SOLV 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR  SOLV 11
      COMMON//A(253000)                                                 SOLV 12
      NRS=NL-1                                                          SOLV 13
      MD=MDM+1                                                          SOLV 14
C     TRIANGULARIZE MATRIX A USING DOOLITTLE METHOD                     SOLV 15
      IPIV=MD-IBW                                                       SOLV 16
      DO 4 IL=1,NRS                                                     SOLV 17
      IPIV=IPIV+IBW                                                     SOLV 18
      PIVOT=A(IPIV)                                                     SOLV 19
      IELIM=IPIV                                                        SOLV 20
      DO 2 IMOD=IL+1,IL+MDM                                             SOLV 21
      IF(IMOD.GT.NL)GOTO3                                               SOLV 22
      IELIM=IELIM+IBW-1                                                 SOLV 23
      C=-A(IELIM)/PIVOT                                                 SOLV 24
C     MODIFY KNOWN VECTOR R                                             SOLV 25
      RL(IMOD)=RL(IMOD)+C*RL(IL)                                        SOLV 26
      DO 1 JJ=1,MDM                                                     SOLV 27
    1 A(IELIM+JJ)=A(IELIM+JJ)+C*A(IPIV+JJ)                              SOLV 28
    2 CONTINUE                                                          SOLV 29
    3 CONTINUE                                                          SOLV 30
    4 CONTINUE                                                          SOLV 31
C     BACK SUBSTITUTE TO SOLVE                                          SOLV 32
      ISOLV=NL*IBW+MD                                                   SOLV 33
      DO 6 IL=NL,1,-1                                                   SOLV 34
      SUM=0.                                                            SOLV 35
      ISOLV=ISOLV-IBW                                                   SOLV 36
      DO 5 I1=1,MDM                                                     SOLV 37
      I2=IL+I1                                                          SOLV 38
      IF(I2.GT.NL)GOTO6                                                 SOLV 39
    5 SUM=SUM+RL(I2)*A(ISOLV+I1)                                        SOLV 40
    6 RL(IL)=(RL(IL)-SUM)/A(ISOLV)                                      SOLV 41
      RETURN                                                            SOLV 42
      END                                                               SOLV 43
```

Attachment 1, continued

```
C     BACK - Back substitutes to determine the remaining unknowns,
C        adds change to old value, and determines largest change.
      SUBROUTINE BACK(IR,JR,KR,NP,V,RL,RU,AU,DU,DELV)                    BACK  1
      LOGICAL LFIN,LHCPT,LVPT                                            BACK  2
      REAL*8 V,RL,RU,AU,DU                                               BACK  3
      REAL KT                                                            BACK  4
      CHARACTER*8 XMESUR,ZMESUR                                          BACK  5
      DIMENSION IR(NEQ),JR(NEQ),KR(NEQ),NP(M,N,L),V(M,N,L),             BACK  6
     c RL(NL),RU(NU),AU(6,NU),DU(NU)                                     BACK  7
      DIMENSION IC(6),JC(6),KC(6)                                        BACK  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,              BACK  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                 BACK 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR   BACK 11
      DATA IC/-1,+1,0,0,0,0/,JC/0,0,-1,+1,0,0/,KC/0,0,0,0,-1,+1/         BACK 12
      DELV=0.                                                            BACK 13
      DO 1 IL=1,NL                                                       BACK 14
      IEQ=IL+NU                                                          BACK 15
      I=IR(IEQ)                                                          BACK 16
      J=JR(IEQ)                                                          BACK 17
      K=KR(IEQ)                                                          BACK 18
      TEST=DABS(RL(IL))                                                  BACK 19
      IF(TEST.GT.ABS(DELV))DELV=RL(IL)                                   BACK 20
    1 V(I,J,K)=V(I,J,K)+RL(IL)                                           BACK 21
      DO 3 IEQ=1,NU                                                      BACK 22
      I=IR(IEQ)                                                          BACK 23
      J=JR(IEQ)                                                          BACK 24
      K=KR(IEQ)                                                          BACK 25
      DO 2 II=1,6                                                        BACK 26
      IIC=I+IC(II)                                                       BACK 27
      IF(IIC.GE.1.AND.IIC.LE.M)THEN                                      BACK 28
           JJC=J+JC(II)                                                  BACK 29
           IF(JJC.GE.1.AND.JJC.LE.N)THEN                                 BACK 30
                KKC=K+KC(II)                                             BACK 31
                IF(KKC.GE.1.AND.KKC.LE.L)THEN                            BACK 32
                     NPIJK=NP(IIC,JJC,KKC)                               BACK 33
                     IF(NPIJK.NE.0)THEN                                  BACK 34
                          IL=NPIJK-NU                                    BACK 35
                          RU(IEQ)=RU(IEQ)-AU(II,IEQ)*RL(IL)              BACK 36
                     ENDIF                                               BACK 37
                ENDIF                                                    BACK 38
           ENDIF                                                         BACK 39
      ENDIF                                                              BACK 40
    2 CONTINUE                                                           BACK 41
      RU(IEQ)=RU(IEQ)/DU(IEQ)                                            BACK 42
      TEST=DABS(RU(IEQ))                                                 BACK 43
      IF(TEST.GT.ABS(DELV))DELV=RU(IEQ)                                  BACK 44
    3 V(I,J,K)=V(I,J,K)+RU(IEQ)                                          BACK 45
      RETURN                                                             BACK 46
      END                                                                BACK 47
```

Attachment 1, continued

```
C     COMPQ - Calculates the interblock ground-water flow.                    COMP  1
      SUBROUTINE COMPQ(IR,JR,KR,IH,HD,HC,DX,DY,DZ,Q)                          COMP  2
      LOGICAL LFIN,LHCPT,LVPT                                                 COMP  3
      REAL KT                                                                 COMP  4
      REAL*8 HD,HIJ                                                           COMP  5
      CHARACTER*8 XMESUR,ZMESUR                                               COMP  6
      DIMENSION IR(NEQ),JR(NEQ),KR(NEQ),IH(M,N,L),                           COMP  7
     c HD(M,N,L),HC(M,N,L),DX(N),DY(M),DZ(L)                                  COMP  8
      DIMENSION Q(NEQ,6)                                                      COMP  9
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDH,MAX,L,M,N,LFIN,MN,                   COMP 10
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                      COMP 11
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR        COMP 12
      DO 1 IEQ=1,NEQ                                                          COMP 13
      I=IR(IEQ)                                                               COMP 14
      J=JR(IEQ)                                                               COMP 15
      K=KR(IEQ)                                                               COMP 16
      IF(IH(I,J,K).EQ.1)THEN                                                  COMP 17
           HIJ=HD(I,J,K)                                                      COMP 18
           HCIJK=HC(I,J,K)                                                    COMP 19
           DZK=DZ(K)                                                          COMP 20
           DXJ=DX(J)                                                          COMP 21
           DYI=DY(I)                                                          COMP 22
           AREAX=2.*DYI*DZK                                                   COMP 23
           AREAY=2.*DXJ*DZK                                                   COMP 24
           AREAZ=2.*DXJ*DYI                                                   COMP 25
           IF((I.EQ.1).OR.(IH(I-1,J,K).EQ.3))THEN                             COMP 26
                Q(IEQ,1)=0.                                                   COMP 27
           ELSE                                                               COMP 28
                Q(IEQ,1)=(HD(I-1,J,K)-HIJ)*HC(I-1,J,K)*HCIJK/                 COMP 29
     c          (HC(I-1,J,K)*DYI+HCIJK*DY(I-1))*AREAY                         COMP 30
           ENDIF                                                              COMP 31
           IF((I.EQ.M).OR.(IH(I+1,J,K).EQ.3))THEN                             COMP 32
                Q(IEQ,2)=0.                                                   COMP 33
           ELSE                                                               COMP 34
                Q(IEQ,2)=(HD(I+1,J,K)-HIJ)*HC(I+1,J,K)*HCIJK/                 COMP 35
     c          (HC(I+1,J,K)*DYI+HCIJK*DY(I+1))*AREAY                         COMP 36
           ENDIF                                                              COMP 37
           IF((J.EQ.1).OR.(IH(I,J-1,K).EQ.3))THEN                             COMP 38
                Q(IEQ,3)=0.                                                   COMP 39
           ELSE                                                               COMP 40
                Q(IEQ,3)=(HD(I,J-1,K)-HIJ)*HC(I,J-1,K)*HCIJK/                 COMP 41
     c          (HC(I,J-1,K)*DXJ+HCIJK*DX(J-1))*AREAX                         COMP 42
           ENDIF                                                              COMP 43
           IF((J.EQ.N).OR.(IH(I,J+1,K).EQ.3))THEN                             COMP 44
                Q(IEQ,4)=0.                                                   COMP 45
           ELSE                                                               COMP 46
                Q(IEQ,4)=(HD(I,J+1,K)-HIJ)*HC(I,J+1,K)*HCIJK/                 COMP 47
     c          (HC(I,J+1,K)*DXJ+HCIJK*DX(J+1))*AREAX                         COMP 48
           ENDIF                                                              COMP 49
           IF((K.EQ.1).OR.(IH(I,J,K-1).EQ.3))THEN                             COMP 50
                Q(IEQ,5)=0.                                                   COMP 51
           ELSE
```

Attachment 1, continued

```
                Q(IEQ,5)=(HD(I,J,K-1)-HIJ)*HC(I,J,K-1)*HCIJK/          COMP 52
     c          (HC(I,J,K-1)*DZK+HCIJK*DZ(K-1))*AREAZ                  COMP 53
           ENDIF                                                       COMP 54
           IF((K.EQ.L).OR.(IH(I,J,K+1).EQ.3))THEN                      COMP 55
                Q(IEQ,6)=0.                                            COMP 56
           ELSE                                                        COMP 57
                Q(IEQ,6)=(HD(I,J,K+1)-HIJ)*HC(I,J,K+1)*HCIJK/          COMP 58
     c          (HC(I,J,K+1)*DZK+HCIJK*DZ(K+1))*AREAZ                  COMP 59
           ENDIF                                                       COMP 60
         ENDIF                                                         COMP 61
       1 CONTINUE                                                      COMP 62
         RETURN                                                        COMP 63
         END                                                           COMP 64
```

Attachment 1, continued

```
C     HEATEQ - Computes coefficient matrix and known vector for
C          heat-flow equations.                                              HEAT  1
      SUBROUTINE HEATEQ(RL,RU,DU,AU,IR,JR,KR,NP,                            HEAT  2
     c TMP,IT,DX,DY,DZ,RCH,DP,HF,Q)                                         HEAT  3
      REAL*8 A,RL,RU,DU,AU,AL,TMP                                           HEAT  4
      REAL*8 REQ,DCOEF,ACOEF,GCOEF,BCOEF,FCOEF,CCOEF,ECOEF,DPART            HEAT  5
      REAL KT,KXY2,KXZ2,KYZ2                                                HEAT  6
      LOGICAL LFIN,LHCPT,LVPT                                               HEAT  7
      CHARACTER*8 XMESUR,ZMESUR                                             HEAT  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,                 HEAT  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                    HEAT 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR      HEAT 11
      COMMON//A(253000)                                                     HEAT 12
      DIMENSION RL(NL),RU(NU),DU(NU),AU(6,NU),AL(6),                        HEAT 13
     c IR(NEQ),JR(NEQ),KR(NEQ),NP(M,N,L),TMP(M,N,L),IT(M,N,L),              HEAT 14
     c DX(N),DY(M),DZ(L),RCH(M,N),DP(M,N),HF(M,N)                           HEAT 15
      DIMENSION IC(6),JC(6),KC(6)                                           HEAT 16
      DIMENSION Q(NEQ,6)                                                    HEAT 17
      DATA IC/-1,+1,0,0,0,0/,JC/0,0,-1,+1,0,0/,KC/0,0,0,0,-1,+1/            HEAT 18
      DO 1 IEQ=1,NU                                                         HEAT 19
      DU(IEQ)=0.                                                            HEAT 20
      RU(IEQ)=0.                                                            HEAT 21
      DO 1 JJ=1,6                                                           HEAT 22
    1 AU(JJ,IEQ)=0.                                                         HEAT 23
      DO 2 JJ=1,6                                                           HEAT 24
    2 AL(JJ)=0.                                                             HEAT 25
      DO 3 IL=1,NL                                                          HEAT 26
    3 RL(IL)=0.                                                             HEAT 27
      IL=0                                                                  HEAT 28
      IA=-IBW                                                               HEAT 29
      MD=MDM+1                                                              HEAT 30
      DO 7 IEQ=1,NEQ                                                        HEAT 31
      IF(IEQ.GT.NU)IL=IL+1                                                  HEAT 32
      I=IR(IEQ)                                                             HEAT 33
      J=JR(IEQ)                                                             HEAT 34
      K=KR(IEQ)                                                             HEAT 35
      DYI=DY(I)                                                             HEAT 36
      DXJ=DX(J)                                                             HEAT 37
      DZK=DZ(K)                                                             HEAT 38
      KXZ2=KT*DXJ*DZK*2.                                                    HEAT 39
      KYZ2=KT*DYI*DZK*2.                                                    HEAT 40
      KXY2=KT*DXJ*DYI*2.                                                    HEAT 41
      DCOEF=0.                                                              HEAT 42
      REQ=0.                                                                HEAT 43
      ACOEF=0.                                                              HEAT 44
      DPART=0.                                                              HEAT 45
      IF(I.GT.1)THEN                                                        HEAT 46
           ITIM1=IT(I-1,J,K)                                                HEAT 47
           IF(ITIM1.NE.3)THEN                                               HEAT 48
                ACOEF=KXZ2/(DY(I-1)+DYI)                                    HEAT 49
                DPART=-ACOEF                                                HEAT 50
           ENDIF
```

Attachment 1, continued

```
      Q1=Q(IEQ,1)                                                    HEAT 51
      IF(Q1.GT.0.)THEN                                               HEAT 52
           ACOEF=ACOEF+SP*Q1                                         HEAT 53
      ELSE IF(Q1.LT.0.)THEN                                          HEAT 54
           DPART=DPART+SP*Q1                                         HEAT 55
      ENDIF                                                          HEAT 56
      DCOEF=DCOEF+DPART                                              HEAT 57
      IF(ITIM1.EQ.1)THEN                                             HEAT 58
           IF(IEQ.GT.NU)THEN                                         HEAT 59
                AL(1)=ACOEF                                          HEAT 60
           ELSE                                                      HEAT 61
                AU(1,IEQ)=ACOEF                                      HEAT 62
           ENDIF                                                     HEAT 63
       ENDIF                                                         HEAT 64
      REQ=REQ-ACOEF*TMP(I-1,J,K)                                     HEAT 65
 ENDIF                                                               HEAT 66
 BCOEF=0.                                                            HEAT 67
 DPART=0.                                                            HEAT 68
 IF(J.NE.1)THEN                                                      HEAT 69
      ITJM1=IT(I,J-1,K)                                              HEAT 70
      IF(ITJM1.NE.3)THEN                                             HEAT 71
           BCOEF=KYZ2/(DX(J-1)+DXJ)                                  HEAT 72
           DPART=-BCOEF                                              HEAT 73
      ENDIF                                                          HEAT 74
      Q3=Q(IEQ,3)                                                    HEAT 75
      IF(Q3.GT.0.)THEN                                               HEAT 76
           BCOEF=BCOEF+SP*Q3                                         HEAT 77
      ELSE IF(Q3.LT.0.)THEN                                          HEAT 78
           DPART=DPART+SP*Q3                                         HEAT 79
      ENDIF                                                          HEAT 80
      DCOEF=DCOEF+DPART                                              HEAT 81
      IF(ITJM1.EQ.1)THEN                                             HEAT 82
           IF(IEQ.GT.NU)THEN                                         HEAT 83
                AL(3)=BCOEF                                          HEAT 84
           ELSE                                                      HEAT 85
                AU(3,IEQ)=BCOEF                                      HEAT 86
           ENDIF                                                     HEAT 87
       ENDIF                                                         HEAT 88
      REQ=REQ-BCOEF*TMP(I,J-1,K)                                     HEAT 89
  ENDIF                                                              HEAT 90
 FCOEF=0.                                                            HEAT 91
 DPART=0.                                                            HEAT 92
 IF(J.LT.N)THEN                                                      HEAT 93
      ITJP1=IT(I,J+1,K)                                              HEAT 94
      IF(ITJP1.NE.3)THEN                                             HEAT 95
           FCOEF=KYZ2/(DXJ+DX(J+1))                                  HEAT 96
           DPART=-FCOEF                                              HEAT 97
      ENDIF                                                          HEAT 98
      Q4=Q(IEQ,4)                                                    HEAT 99
      IF(Q4.GT.0.)THEN                                               HEAT100
           FCOEF=FCOEF+SP*Q4                                         HEAT101
      ELSE IF(Q4.LT.0.)THEN                                          HEAT102
```

Attachment 1, continued

```
            DPART=DPART+SP*Q4                                    HEAT103
          ENDIF                                                  HEAT104
          DCOEF=DCOEF+DPART                                      HEAT105
          IF(ITJP1.EQ.1)THEN                                     HEAT106
               IF(IEQ.GT.NU)THEN                                 HEAT107
                    AL(4)=FCOEF                                  HEAT108
               ELSE                                              HEAT109
                    AU(4,IEQ)=FCOEF                              HEAT110
               ENDIF                                             HEAT111
           ENDIF                                                 HEAT112
          REQ=REQ-FCOEF*TMP(I,J+1,K)                             HEAT113
      ENDIF                                                      HEAT114
     GCOEF=0.                                                    HEAT115
     DPART=0.                                                    HEAT116
     IF(I.LT.M)THEN                                              HEAT117
          ITIP1=IT(I+1,J,K)                                      HEAT118
          IF(ITIP1.NE.3)THEN                                     HEAT119
               GCOEF=KXZ2/(DXI+DX(I+1))                          HEAT120
               DPART=-GCOEF                                      HEAT121
          ENDIF                                                  HEAT122
          Q2=Q(IEQ,2)                                            HEAT123
          IF(Q2.GT.0.)THEN                                       HEAT124
               GCOEF=GCOEF+SP*Q2                                 HEAT125
          ELSE IF(Q2.LT.0.)THEN                                  HEAT126
               DPART=DPART+SP*Q2                                 HEAT127
          ENDIF                                                  HEAT128
          DCOEF=DCOEF+DPART                                      HEAT129
          IF(ITIP1.EQ.1)THEN                                     HEAT130
               IF(IEQ.GT.NU)THEN                                 HEAT131
                    AL(2)=GCOEF                                  HEAT132
               ELSE                                              HEAT133
                    AU(2,IEQ)=GCOEF                              HEAT134
               ENDIF                                             HEAT135
           ENDIF                                                 HEAT136
          REQ=REQ-GCOEF*TMP(I+1,J,K)                             HEAT137
      ENDIF                                                      HEAT138
     IF(K.EQ.1)THEN                                              HEAT139
          DCOEF=DCOEF-DP(I,J)                                    HEAT140
          REQ=REQ-DP(I,J)*TTOP-SP*TEMPR*RCH(I,J)                 HEAT141
     ELSE                                                        HEAT142
          CCOEF=0.                                               HEAT143
          DPART=0.                                               HEAT144
          ITKM1=IT(I,J,K-1)                                      HEAT145
          IF(ITKM1.NE.1)THEN                                     HEAT146
               DCOEF=DCOEF-DP(I,J)                               HEAT147
               REQ=REQ-DP(I,J)*TTOP-SP*TEMPR*RCH(I,J)            HEAT148
          ENDIF                                                  HEAT149
          IF(ITKM1.NE.3)THEN                                     HEAT150
               CCOEF=KXY2/(DZK+DZ(K-1))                          HEAT151
               DPART=-CCOEF                                      HEAT152
          ENDIF                                                  HEAT153
          Q5=Q(IEQ,5)                                            HEAT154
```

Attachment 1, continued

```
      IF(Q5.GT.0.)THEN                                          HEAT155
           CCOEF=CCOEF+SP*Q5                                    HEAT156
      ELSE IF(Q5.LT.0.)THEN                                     HEAT157
           DPART=DPART+SP*Q5                                    HEAT158
      ENDIF                                                     HEAT159
      DCOEF=DCOEF+DPART                                         HEAT160
      IF(IEQ.LE.NU)THEN                                         HEAT161
           AU(5,IEQ)=CCOEF                                      HEAT162
      ELSE                                                      HEAT163
           AL(5)=CCOEF                                          HEAT164
      ENDIF                                                     HEAT165
      REQ=REQ-CCOEF*TMP(I,J,K-1)                                HEAT166
 ENDIF                                                          HEAT167
 IF(K.EQ.L)THEN                                                 HEAT168
      REQ=REQ-HF(I,J)                                           HEAT169
 ELSE                                                           HEAT170
      ITKP1=IT(I,J,K+1)                                         HEAT171
      ECOEF=0.                                                  HEAT172
      DPART=0.                                                  HEAT173
      IF(ITKP1.NE.3)THEN                                        HEAT174
           ECOEF=KXY2/(DZK+DZ(K+1))                             HEAT175
           DPART=-ECOEF                                         HEAT176
      ENDIF                                                     HEAT177
      Q6=Q(IEQ,6)                                               HEAT178
      IF(Q6.GT.0.)THEN                                          HEAT179
           ECOEF=ECOEF+SP*Q6                                    HEAT180
      ELSE IF(Q6.LT.0.)THEN                                     HEAT181
           DPART=DPART+SP*Q6                                    HEAT182
      ENDIF                                                     HEAT183
           DCOEF=DCOEF+DPART                                    HEAT184
      IF(ITKP1.EQ.1)THEN                                        HEAT185
           IF(IEQ.GT.NU)THEN                                    HEAT186
                AL(6)=ECOEF                                     HEAT187
           ELSE                                                 HEAT188
                AU(6,IEQ)=ECOEF                                 HEAT189
           ENDIF                                                HEAT190
      ENDIF                                                     HEAT191
      REQ=REQ-ECOEF*TMP(I,J,K+1)                                HEAT192
 ENDIF                                                          HEAT193
 IF(IEQ.LE.NU)THEN                                              HEAT194
      RU(IEQ)=REQ-DCOEF*TMP(I,J,K)                              HEAT195
      DU(IEQ)=DCOEF                                             HEAT196
 ELSE                                                           HEAT197
      RL(IL)=REQ-DCOEF*TMP(I,J,K)                               HEAT198
      IA=IA+IBW                                                 HEAT199
      DO 4 ISUB=IA+1,IA+IBW                                     HEAT200
4     A(ISUB)=0.                                                HEAT201
      A(IA+MD)=DCOEF                                            HEAT202
      DO 6 II=1,6                                               HEAT203
      IXII=I+IC(II)                                             HEAT204
      IF((IXII.LE.0).OR.(IXII.GT.M))GOTO6                       HEAT205
      JXII=J+JC(II)                                             HEAT206
```

Attachment 1, continued

```
      IF((JXII.LE.0).OR.(JXII.GT.N))GOTO6                              HEAT207
      KXII=K+KC(II)                                                    HEAT208
      IF((KXII.LE.0).OR.(KXII.GT.L))GOTO6                              HEAT209
      NPQ=NP(IXII,JXII,KXII)                                           HEAT210
      IF(NPQ.EQ.0)GOTO6                                                HEAT211
      DO 5 JJ=1,6                                                      HEAT212
      IXJJ=I+IC(JJ)+IC(II)                                             HEAT213
      IF((IXJJ.LE.0).OR.(IXJJ.GT.M))GOTO5                              HEAT214
      JXJJ=J+JC(JJ)+JC(II)                                             HEAT215
      IF((JXJJ.LE.0).OR.(JXJJ.GT.N))GOTO5                              HEAT216
      KXJJ=K+KC(JJ)+KC(II)                                             HEAT217
      IF((KXJJ.LE.0).OR.(KXJJ.GT.L))GOTO5                              HEAT218
      NPP=NP(IXJJ,JXJJ,KXJJ)                                           HEAT219
      IF(NPP.EQ.0)GOTO5                                                HEAT220
      NN=IA+MD-IEQ+NPP                                                 HEAT221
      A(NN)=A(NN)-AU(JJ,NPQ)*AL(II)/DU(NPQ)                            HEAT222
5     CONTINUE                                                         HEAT223
      RL(IL)=RL(IL)-RU(NPQ)*AL(II)/DU(NPQ)                             HEAT224
6     CONTINUE                                                         HEAT225
    ENDIF                                                              HEAT226
7   CONTINUE                                                           HEAT227
    RETURN                                                             HEAT228
    END                                                                HEAT229
```

Attachment 1, continued

```
C     BAL - Computes water and heat balances.                                    BAL  1
      SUBROUTINE BAL(TMP,HC,DX,DY,DZ,IT,HF,DP,RCH,IH,HD)                         BAL  2
      LOGICAL LFIN,LHCPT,LVPT                                                    BAL  3
      REAL KT,KXY,KXZ,KYZ                                                        BAL  4
      REAL*8 HD,TMP                                                              BAL  5
      CHARACTER*8 XMESUR,ZMESUR                                                  BAL  6
      DIMENSION TMP(M,N,L),HC(M,N,L),DX(N),DY(M),DZ(L),IT(M,N,L),                BAL  7
     c HF(M,N),DP(M,N),RCH(M,N),IH(M,N,L),HD(M,N,L)                              BAL  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,                      BAL  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                         BAL 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR           BAL 11
      CTO=.0                                                                     BAL 12
      CTI=0.                                                                     BAL 13
      CFO=0.                                                                     BAL 14
      CFI=0.                                                                     BAL 15
      CVO=0.                                                                     BAL 16
      CVI=0.                                                                     BAL 17
      CHO=0.                                                                     BAL 18
      CHI=0.                                                                     BAL 19
      CSO=0.                                                                     BAL 20
      CSI=0.                                                                     BAL 21
      RECH=0.                                                                    BAL 22
      DO 1 K=1,L                                                                 BAL 23
      DO 1 J=1,N                                                                 BAL 24
      DO 1 I=1,M                                                                 BAL 25
      IF(IT(I,J,K).EQ.1)THEN                                                     BAL 26
           IF(I.NE.1)THEN                                                        BAL 27
                IF(IT(I-1,J,K).EQ.2)THEN                                         BAL 28
                     CT=(TMP(I,J,K)-TMP(I-1,J,K))*KT*DX(J)*DZ(K)*2.              BAL 29
     c               /(DY(I)+DY(I-1))                                            BAL 30
                     IF(CT.LT.0.)THEN                                            BAL 31
                          CTI=CTI-CT                                             BAL 32
                     ELSE IF(CT.GT.0.)THEN                                       BAL 33
                          CTO=CTO+CT                                             BAL 34
                     ENDIF                                                       BAL 35
                ENDIF                                                            BAL 36
                IF(IH(I-1,J,K).EQ.2)THEN                                         BAL 37
                     CH=(HD(I,J,K)-HD(I-1,J,K))*HC(I-1,J,K)*HC(I,J,K)            BAL 38
     c               *2.*DX(J)*DZ(K)                                             BAL 39
     c               /(HC(I-1,J,K)*DY(I)+HC(I,J,K)*DY(I-1))                      BAL 40
                     IF(CH.LT.0.)THEN                                            BAL 41
                          CHI=CHI-CH                                             BAL 42
                          CVI=CVI-CH*SP*TMP(I-1,J,K)                             BAL 43
                     ELSE IF(CH.GT.0.)THEN                                       BAL 44
                          CHO=CHO+CH                                             BAL 45
                          CVO=CVO+CH*SP*TMP(I,J,K)                               BAL 46
                     ENDIF                                                       BAL 47
                ENDIF                                                            BAL 48
           ENDIF                                                                 BAL 49
           IF(I.NE.M)THEN                                                        BAL 50
                IF(IT(I+1,J,K).EQ.2)THEN                                         BAL 51
                     CT=(TMP(I,J,K)-TMP(I+1,J,K))*KT*DX(J)*DZ(K)*2.
```

Attachment 1, continued

```
c                /(DY(I)+DY(I+1))                                          BAL 52
                 IF(CT.LT.0.)THEN                                          BAL 53
                      CTI=CTI-CT                                           BAL 54
                 ELSE IF(CT.GT.0.)THEN                                     BAL 55
                      CTO=CTO+CT                                           BAL 56
                 ENDIF                                                     BAL 57
               ENDIF                                                       BAL 58
               IF(IH(I+1,J,K).EQ.2)THEN                                    BAL 59
                 CH=(HD(I,J,K)-HD(I+1,J,K))*HC(I+1,J,K)*HC(I,J,K)          BAL 60
                 *2.*DX(J)*DZ(K)                                           BAL 61
c                /(HC(I+1,J,K)*DY(I)+HC(I,J,K)*DY(I+1))                    BAL 62
c                IF(CH.LT.0.)THEN                                          BAL 63
                      CHI=CHI-CH                                           BAL 64
                      CVI=CVI-CH*SP*TMP(I+1,J,K)                           BAL 65
                 ELSE IF(CH.GT.0.)THEN                                     BAL 66
                      CHO=CHO+CH                                           BAL 67
                      CVO=CVO+CH*SP*TMP(I,J,K)                             BAL 68
                 ENDIF                                                     BAL 69
               ENDIF                                                       BAL 70
             ENDIF                                                         BAL 71
             IF(J.NE.1)THEN                                                BAL 72
               IF(IT(I,J-1,K).EQ.2)THEN                                    BAL 73
                 CT=(TMP(I,J,K)-TMP(I,J-1,K))*KT*DY(I)*DZ(K)*2.            BAL 74
                 /(DX(J)+DX(J-1))                                          BAL 75
c                IF(CT.LT.0.)THEN                                          BAL 76
                      CTI=CTI-CT                                           BAL 77
                 ELSE IF(CT.GT.0.)THEN                                     BAL 78
                      CTO=CTO+CT                                           BAL 79
                 ENDIF                                                     BAL 80
               ENDIF                                                       BAL 81
               IF(IH(I,J-1,K).EQ.2)THEN                                    BAL 82
                 CH=(HD(I,J,K)-HD(I,J-1,K))*HC(I,J-1,K)*HC(I,J,K)          BAL 83
                 *2.*DY(I)*DZ(K)                                           BAL 84
c                /(HC(I,J-1,K)*DX(J)+HC(I,J,K)*DX(J-1))                    BAL 85
c                IF(CH.LT.0.)THEN                                          BAL 86
                      CHI=CHI-CH                                           BAL 87
                      CVI=CVI-CH*SP*TMP(I,J-1,K)                           BAL 88
                 ELSE IF(CH.GT.0.)THEN                                     BAL 89
                      CHO=CHO+CH                                           BAL 90
                      CVO=CVO+CH*SP*TMP(I,J,K)                             BAL 91
                 ENDIF                                                     BAL 92
               ENDIF                                                       BAL 93
             ENDIF                                                         BAL 94
             IF(J.NE.N)THEN                                                BAL 95
               IF(IT(I,J+1,K).EQ.2)THEN                                    BAL 96
                 CT=(TMP(I,J,K)-TMP(I,J+1,K))*KT*DY(I)*DZ(K)*2.            BAL 97
                 /(DX(J)+DX(J+1))                                          BAL 98
c                IF(CT.LT.0.)THEN                                          BAL 99
                      CTI=CTI-CT                                           BAL100
                 ELSE IF(CT.GT.0.)THEN                                     BAL101
                      CTO=CTO+CT                                           BAL102
                 ENDIF                                                     BAL103
```

```
      ENDIF                                                             BAL104
      IF(IH(I,J+1,K).EQ.2)THEN                                          BAL105
```

Attachment 1, continued

```
                CH=(HD(I,J,K)-HD(I,J+1,K))*HC(I,J+1,K)*HC(I,J,K)      BAL106
                *2.*DY(I)*DZ(K)                                       BAL107
c               /(HC(I,J+1,K)*DX(J)+HC(I,J,K)*DX(J+1))                BAL108
c               IF(CH.LT.0.)THEN                                      BAL109
                     CHI=CHI-CH                                       BAL110
                     CVI=CVI-CH*SP*TMP(I,J+1,K)                       BAL111
                ELSE IF(CH.GT.0.)THEN                                 BAL112
                     CHO=CHO+CH                                       BAL113
                     CVO=CVO+CH*SP*TMP(I,J,K)                         BAL114
                ENDIF                                                 BAL115
           ENDIF                                                      BAL116
      ENDIF                                                           BAL117
      IF(K.EQ.1.OR.IT(I,J,K-1).NE.1)THEN                              BAL118
           CS=(TTOP-TMP(I,J,K))*DP(I,J)                               BAL119
           IF(CS.LT.0.)THEN                                           BAL120
                CSO=CSO-CS                                            BAL121
           ELSE IF(CS.GT.0.)THEN                                      BAL122
                CSI=CSI+CS                                            BAL123
           ENDIF                                                      BAL124
           RECH=RECH+RCH(I,J)                                         BAL125
           CVI=CVI+SP*TEMPR*RCH(I,J)                                  BAL126
      ENDIF                                                           BAL127
      IF(K.NE.1)THEN                                                  BAL128
           IF(IT(I,J,K-1).EQ.2)THEN                                   BAL129
                CT=(TMP(I,J,K)-TMP(I,J,K-1))*KT*DX(J)*DY(I)*2.        BAL130
                /(DZ(K)+DZ(K-1))                                      BAL131
c               IF(CT.LT.0.)THEN                                      BAL132
                     CTI=CTI-CT                                       BAL133
                ELSE IF(CT.GT.0.)THEN                                 BAL134
                     CTO=CTO+CT                                       BAL135
                ENDIF                                                 BAL136
           ENDIF                                                      BAL137
           IF(IH(I,J,K-1).EQ.2)THEN                                   BAL138
                CH=(HD(I,J,K)-HD(I,J,K-1))*HC(I,J,K-1)*HC(I,J,K)      BAL139
                *2.*DX(J)*DY(I)                                       BAL140
c               /(HC(I,J,K-1)*DZ(K)+HC(I,J,K)*DZ(K-1))                BAL141
c               IF(CH.LT.0.)THEN                                      BAL142
                     CHI=CHI-CH                                       BAL143
                     CVI=CVI-CH*SP*TMP(I,J,K-1)                       BAL144
                ELSE IF(CH.GT.0.)THEN                                 BAL145
                     CHO=CHO+CH                                       BAL146
                     CVO=CVO+CH*SP*TMP(I,J,K)                         BAL147
                ENDIF                                                 BAL148
           ENDIF                                                      BAL149
      ENDIF                                                           BAL150
      IF(K.EQ.L)THEN                                                  BAL151
           IF(HF(I,J).LT.0.)THEN                                      BAL152
                CFO=CFO-HF(I,J)                                       BAL153
           ELSE IF(HF(I,J).GT.0.)THEN                                 BAL154
                CFI=CFI+HF(I,J)                                       BAL155
           ENDIF                                                      BAL156
      ELSE                                                            BAL157
```

Attachment 1, continued

```
            IF(IT(I,J,K+1).EQ.2)THEN                                    BAL158
                 CT=(TMP(I,J,K)-TMP(I,J,K+1))*KT*DX(J)*DY(I)*2.         BAL159
     c           /(DZ(K)+DZ(K+1))                                       BAL160
                 IF(CT.LT.0.)THEN                                       BAL161
                      CTI=CTI-CT                                        BAL162
                 ELSE IF(CT.GT.0.)THEN                                  BAL163
                      CTO=CTO+CT                                        BAL164
                 ENDIF                                                  BAL165
            ENDIF                                                       BAL166
            IF(IH(I,J,K+1).EQ.2)THEN                                    BAL167
                 CH=(HD(I,J,K)-HD(I,J,K+1))*HC(I,J,K+1)*HC(I,J,K)       BAL168
     c           *2.*DX(J)*DY(I)                                        BAL169
     c           /(HC(I,J,K+1)*DZ(K)+HC(I,J,K)*DZ(K+1))                 BAL170
                 IF(CH.LT.0.)THEN                                       BAL171
                      CHI=CHI-CH                                        BAL172
                      CVI=CVI-CH*SP*TMP(I,J,K+1)                        BAL173
                 ELSE IF(CH.GT.0.)THEN                                  BAL174
                      CHO=CHO+CH                                        BAL175
                      CVO=CVO+CH*SP*TMP(I,J,K)                          BAL176
                 ENDIF                                                  BAL177
            ENDIF                                                       BAL178
       ENDIF                                                            BAL179
  ENDIF                                                                 BAL180
1 CONTINUE                                                              BAL181
  HBAL=0.                                                               BAL182
  DENOM=CVO+CVI+CSO+CSI+CFO+CFI+CTO+CTI                                 BAL183
  HNUM=CVI+CSI+CFI+CTI-(CVO+CSO+CFO+CTO)                                BAL184
  IF(DENOM.NE.0.)HBAL=100.*HNUM/DENOM                                   BAL185
  WBAL=0.                                                               BAL186
  DENOM=CHO+CHI+RECH                                                    BAL187
  WNUM=CHI+RECH-CHO                                                     BAL188
  IF(DENOM.NE.0.)WBAL=100.*WNUM/DENOM                                   BAL189
  WRITE(6,11)CVI,CVO,CFI,CFO,CSI,CSO,CTI,CTO,HNUM,HBAL,                BAL190
 c RECH,CHI,CHO,WNUM,WBAL                                               BAL191
  WRITE(6,7)                                                            BAL192
  DO 2 K=1,L                                                            BAL193
  WRITE(6,6)K                                                           BAL194
  DO 2 I=1,M                                                            BAL195
2 WRITE(6,9) (TMP(I,J,K),J=1,N)                                         BAL196
  WRITE(6,8)                                                            BAL197
  DO 3 K=1,L                                                            BAL198
  WRITE(6,6)K                                                           BAL199
  DO 3 I=1,M                                                            BAL200
3 WRITE(6,9)(HD(I,J,K),J=1,N)                                           BAL201
  IF(.NOT.LHCPT)GOTO5                                                   BAL202
  WRITE(6,10)                                                           BAL203
  DO 4 K=1,L                                                            BAL204
  WRITE(6,6)K                                                           BAL205
  DO 4 I=1,M                                                            BAL206
4 WRITE(6,9)(HC(I,J,K),J=1,N)                                           BAL207
5 CONTINUE                                                              BAL208
  RETURN                                                                BAL209
```

Attachement 1, continued

```
    6 FORMAT(/1X,'LAYER ',I2/1X,8(1H-))                                    BAL210
    7 FORMAT(/1X,'TEMPERATURE, DEG. F'/1X,19(1H-))                         BAL211
    8 FORMAT(/1X,'HEAD, FEET'/1X,10(1H-))                                  BAL212
    9 FORMAT(1X,1P10E12.4)                                                 BAL213
   10 FORMAT(/1X,'HYD. COND. (FT./DAY) AT COMP. TEMP.'/1X,35(1H-))         BAL214
   11 FORMAT(/13X,'STRESS COMPONENTS FOR HEAT, UNITS = BTU/DAY'/           BAL215
     c 13X,43(1H-)/                                                       BAL216
     c 33X,'IN',11X,'OUT'/19X,'ADVECTION = ',1PE12.4,1X,E12.4/            BAL217
     c 8X,'HEAT FLOW FROM BELOW = ',E12.4,1X,E12.4/                       BAL218
     c 8X,'HEAT FLOW TO SURFACE = ',E12.4,1X,E12.4/                       BAL219
     c 3X,'CONSTANT TEMP. BOUNDARIES = ',E12.4,1X,E12.4/                  BAL220
     c 22X,'IN-OUT = ',E12.4/14X,'HEAT IMBALANCE = ',0PF7.2,' PERCENT'/   BAL221
     c /12X,'STRESS COMPONENTS FOR WATER, UNITS = CUBIC FT./DAY'/         BAL222
     c 12X,50(1H-)/                                                       BAL223
     c 33X,'IN',11X,'OUT'/                                                BAL224
     c 20X,'RECHARGE = ',1PE12.4/                                         BAL225
     c 4X,'CONSTANT HEAD BOUNDARIES = ',E12.4,1X,E12.4/                   BAL226
     c 22X,'IN-OUT = ',E12.4/13X,'WATER IMBALANCE = ',0PF7.2,' PERCENT')  BAL227
      END                                                                 BAL228
```

Attachment 1, continued

```
C     VELO - Determines velocities.
      SUBROUTINE VELO(IH,HD,HC,DX,DY,DZ,                                VELO  1
     c VX1,VX2,VY1,VY2,VZ1,VZ2,POR,RCH)                                 VELO  2
      LOGICAL LFIN,LHCPT,LVPT                                           VELO  3
      REAL KT                                                           VELO  4
      REAL*8 HD,HIJK                                                    VELO  5
      CHARACTER*8 XMESUR,ZMESUR                                         VELO  6
      DIMENSION IH(M,M,L),HD(M,M,L),HC(M,M,L),                          VELO  7
     c DX(M),DY(M),DZ(L),VX1(M,M,L),VX2(M,M,L),VY1(M,M,L),              VELO  8
     c VY2(M,M,L),VZ1(M,M,L),VZ2(M,M,L),POR(M,M,L),RCH(M,M)             VELO  9
      COMMON/VAR/MU,NL,IBW,MEQ,IGATE,MDM,MAX,L,M,M,LFIN,MM,             VELO 10
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                VELO 11
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR  VELO 12
      DO 1 K=1,L                                                        VELO 13
      DO 1 J=1,M                                                        VELO 14
      DO 1 I=1,M                                                        VELO 15
      IHIJK=IH(I,J,K)                                                   VELO 16
      IF(IHIJK.EQ.1)THEN                                                VELO 17
           HIJK=HD(I,J,K)                                               VELO 18
           HCIJK=HC(I,J,K)                                              VELO 19
           PORIJK=POR(I,J,K)                                            VELO 20
           IF((J.EQ.1).OR.(IH(I,J-1,K).EQ.3))THEN                       VELO 21
                VX1(I,J,K)=0.                                           VELO 22
           ELSE                                                         VELO 23
                VX1(I,J,K)=2.*HCIJK*HC(I,J-1,K)/(HCIJK*DX(J-1)          VELO 24
     c          +HC(I,J-1,K)*DX(J))*                                    VELO 25
     c          (HD(I,J-1,K)-HIJK)/PORIJK                               VELO 26
           ENDIF                                                        VELO 27
           IF((J.EQ.M).OR.(IH(I,J+1,K).EQ.3))THEN                       VELO 28
                VX2(I,J,K)=0.                                           VELO 29
           ELSE                                                         VELO 30
                VX2(I,J,K)=2.*HCIJK*HC(I,J+1,K)/(HCIJK*DX(J+1)          VELO 31
     c          +HC(I,J+1,K)*DX(J))*                                    VELO 32
     c          (HIJK-HD(I,J+1,K))/PORIJK                               VELO 33
           ENDIF                                                        VELO 34
           IF((I.EQ.1).OR.(IH(I-1,J,K).EQ.3))THEN                       VELO 35
                VY1(I,J,K)=0.                                           VELO 36
           ELSE                                                         VELO 37
                VY1(I,J,K)=2.*HCIJK*HC(I-1,J,K)/(HCIJK*DY(I-1)          VELO 38
     c          +HC(I-1,J,K)*DY(I))*                                    VELO 39
     c          (HD(I-1,J,K)-HIJK)/PORIJK                               VELO 40
           ENDIF                                                        VELO 41
           IF((I.EQ.M).OR.(IH(I+1,J,K).EQ.3))THEN                       VELO 42
                VY2(I,J,K)=0.                                           VELO 43
           ELSE                                                         VELO 44
                VY2(I,J,K)=2.*HCIJK*HC(I+1,J,K)/(HCIJK*DY(I+1)          VELO 45
     c          +HC(I+1,J,K)*DY(I))*                                    VELO 46
     c          (HIJK-HD(I+1,J,K))/PORIJK                               VELO 47
           ENDIF                                                        VELO 48
           IF((K.EQ.1).OR.(IH(I,J,K-1).EQ.3))THEN                       VELO 49
                VZ1(I,J,K)=RCH(I,J)/(DX(J)*DY(I)*PORIJK)                VELO 50
           ELSE                                                         VELO 51
```

Attachment 1, continued

```
                VZ1(I,J,K)=2.*HCIJK*HC(I,J,K-1)/(HCIJK*DZ(K-1)          VELO 52
                                                                        VELO 53
     c          +HC(I,J,K-1)*DZ(K))*                                    VELO 54
     c          (HD(I,J,K-1)-HIJK)/PORIJK                               VELO 55
           ENDIF                                                        VELO 56
           IF((K.EQ.L).OR.(IH(I,J,K+1).EQ.3))THEN                       VELO 57
                VZ2(I,J,K)=0.                                           VELO 58
           ELSE                                                         VELO 59
                VZ2(I,J,K)=2.*HCIJK*HC(I,J,K+1)/(HCIJK*DZ(K+1)          VELO 60
     c          +HC(I,J,K+1)*DZ(K))*                                    VELO 61
     c          (HIJK-HD(I,J,K+1))/PORIJK                               VELO 62
           ENDIF                                                        VELO 63
         ELSE                                                           VELO 64
           VX1(I,J,K)=0.                                                VELO 65
           VX2(I,J,K)=0.                                                VELO 66
           VY1(I,J,K)=0.                                                VELO 67
           VY2(I,J,K)=0.                                                VELO 68
           VZ1(I,J,K)=0.                                                VELO 69
           VZ2(I,J,K)=0.                                                VELO 70
         ENDIF                                                          VELO 71
       1 CONTINUE                                                       VELO 72
         IF(LVPT)THEN                                                   VELO 73
           WRITE(6,5)                                                   VELO 74
           DO 2 K=1,L                                                   VELO 75
           WRITE(6,6)K                                                  VELO 76
           DO 2 I=1,M                                                   VELO 77
       2   WRITE(6,7)(VX1(I,J,K),VX2(I,J,K),J=1,N)                      VELO 78
           WRITE(6,8)                                                   VELO 79
           DO 3 K=1,L                                                   VELO 80
           WRITE(6,6)K                                                  VELO 81
           DO 3 I=1,M                                                   VELO 82
       3   WRITE(6,7)(VY1(I,J,K),VY2(I,J,K),J=1,N)                      VELO 83
           WRITE(6,9)                                                   VELO 84
           DO 4 K=1,L                                                   VELO 85
           WRITE(6,6)K                                                  VELO 86
           DO 4 I=1,M                                                   VELO 87
       4   WRITE(6,7)(VZ1(I,J,K),VZ2(I,J,K),J=1,N)                      VELO 88
         ENDIF                                                          VELO 89
         RETURN                                                         VELO 90
       5 FORMAT (1H1,21HX VELOCITIES, FT./DAY)                          VELO 91
       6 FORMAT(1X,'LAYER ',I2/1X,8(1H-))                               VELO 92
       7 FORMAT (1H ,8G12.3)                                            VELO 93
       8 FORMAT (1H1,21HY VELOCITIES, FT./DAY)                          VELO 94
       9 FORMAT (1H1,21HZ VELOCITIES, FT./DAY)                          VELO 95
         END
```

Attachment 1, continued

```
C     MOVE - Determines location of discharge and travel time to
C        discharge point for each active node.
      SUBROUTINE MOVE(IH,VX1,VX2,VY1,VY2,VZ1,VZ2,DX,DY,DZ,TIM,IDS)
      LOGICAL LFIN,LHCPT,LVPT,LPATH
      REAL KT
      CHARACTER*8 XMESUR,ZMESUR
      DIMENSION IH(M,N,L),VX1(M,N,L),VX2(M,N,L),VY1(M,N,L),
     c VY2(M,N,L),VZ1(M,N,L),VZ2(M,N,L),
     c DX(M),DY(M),DZ(L),TIM(M,N,L),IDS(M,N,L)
      COMMON/VAR/MU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR
      DATA LPATH/.FALSE./
      DO 1 K=1,L
      DO 1 J=1,N
      DO 1 I=1,M
      IDSYXZ=0
      TISUM=0.
      IF(IH(I,J,K).EQ.1)THEN
           INY=I
           INX=J
           INZ=K
           XO=DX(J)/2.
           YO=DY(I)/2.
           ZO=DZ(K)/2.
           CALL MSUB(XO,YO,ZO,INY,INX,INZ,IDSYXZ,VX1,VX2,
     c      VY1,VY2,VZ1,VZ2,DX,DY,DZ,IH,TISUM,LPATH,
     c      DISTX,DISTZ,NREC,NRLIM,PATH,IPA,KSYM,XSF,YSF)
      ENDIF
      TISUM=TISUM/365.25
      TIM(I,J,K)=TISUM
      IDS(I,J,K)=IDSYXZ
    1 CONTINUE
      J1=1
      J2=12
    2 IF(J2.GT.N)J2=N
      WRITE(6,8)
      IF(M.EQ.1)WRITE(6,9)(J,J=J1,J2)
      DO 3 K=1,L
      WRITE(6,10)K
      IF(M.GT.1)WRITE(6,9)(J,J=J1,J2)
      DO 3 I=1,M
    3 WRITE(6,11)I,(TIM(I,J,K),J=J1,J2)
      J1=J1+12
      IF(J1.GT.N)GOTO4
      J2=J2+12
      GOTO2
    4 J1=1
      J2=12
    5 IF(J2.GT.N)J2=N
      WRITE(6,12)
      IF(M.EQ.1)WRITE(6,9)(J,J=J1,J2)
```

Attachment 1, continued

```
      DO  6 K=1,L                                                    MOVE 51
      WRITE(6,10)K                                                   MOVE 52
      IF(N.GT.1)WRITE(6,9)(J,J=J1,J2)                                MOVE 53
      DO  6 I=1,M                                                    MOVE 54
    6 WRITE(6,13)I,(IDS(I,J,K),J=J1,J2)                              MOVE 55
      J1=J1+12                                                       MOVE 56
      IF(J1.GT.N)GOTO7                                               MOVE 57
      J2=J2+12                                                       MOVE 58
      GOTO5                                                          MOVE 59
    7 CONTINUE                                                       MOVE 60
      RETURN                                                         MOVE 61
    8 FORMAT(1H1,'TIME OF TRAVEL TO DISCHARGE POINT, YEARS')         MOVE 62
    9 FORMAT(6X,12I10)                                               MOVE 63
   10 FORMAT(1X,'LAYER ',I2/1X,8(1H-))                               MOVE 64
   11 FORMAT(1H ,I3,2X,1P12E10.2)                                    MOVE 65
   12 FORMAT(1H1,'DISCHARGE POINT:Y,X,Z')                            MOVE 66
   13 FORMAT(1H ,I3,2X,12I10)                                        MOVE 67
      END                                                            MOVE 68
```

Attachment 1, continued

```
C     MSUB - Moves a water particle from initial point to discharge.
      SUBROUTINE MSUB(X0,Y0,Z0,I,J,K,IDSYXZ,VX1,VX2,                    MSUB  1
     c VY1,VY2,VZ1,VZ2,DX,DY,DZ,IH,TISUM,LPATH,                         MSUB  2
     c DISTX,DISTZ,NREC,NRLIM,PATH,IPATH,KSYM,XSF,ZSF)                  MSUB  3
      LOGICAL LFIN,LHCPT,LVPT,LPATH                                     MSUB  4
      REAL KT                                                           MSUB  5
      CHARACTER*8 XMESUR,ZMESUR                                         MSUB  6
      DIMENSION IH(M,N,L),VX1(M,N,L),VX2(M,N,L),VY1(M,N,L),             MSUB  7
     c VY2(M,N,L),VZ1(M,N,L),VZ2(M,N,L),                               MSUB  8
     c DX(N),DY(M),DZ(L),DISTX(N),DISTZ(L)                              MSUB  9
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,             MSUB 10
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                MSUB 11
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR  MSUB 12
      IF(LPATH)THEN                                                     MSUB 13
                                                                        MSUB 14
           IF(KSYM.GE.50)THEN                                           MSUB 15
                KSYM=1                                                  MSUB 16
           ELSE                                                         MSUB 17
                KSYM=KSYM+1                                             MSUB 18
           ENDIF                                                        MSUB 19
      ENDIF                                                             MSUB 20
      X=X0                                                              MSUB 21
      Y=Y0                                                              MSUB 22
      Z=Z0                                                              MSUB 23
      IORG=I                                                            MSUB 24
      JORG=J                                                            MSUB 25
      KORG=K                                                            MSUB 26
      LIMIT=30*(N+M+L)                                                  MSUB 27
      NMOVE=0                                                           MSUB 28
      TISUM=0.                                                          MSUB 29
      TX=1.E+38                                                         MSUB 30
      TY=1.E+38                                                         MSUB 31
      TZ=1.E+38                                                         MSUB 32
      IF(LPATH)THEN                                                     MSUB 33
           WRITE(6,2)J,K,X0,Z0,TISUM                                    MSUB 34
           CALL PSUB(DISTX,DX,XSF,DISTZ,ZSF,DZ,                         MSUB 35
     c          IPATH,PATH,NREC,NRLIM,KSYM,X0,Z0,J,K)                  MSUB 36
      ENDIF                                                             MSUB 37
    1 DXJ=DX(J)                                                         MSUB 38
      DYI=DY(I)                                                         MSUB 39
      DZK=DZ(K)                                                         MSUB 40
      IF(N.GT.1)THEN                                                    MSUB 41
           VX01=VX1(I,J,K)                                              MSUB 42
           VX02=VX2(I,J,K)                                              MSUB 43
           CALL   MSUB1(X0,DXJ,VX0,VX01,VX02,BX,TX,J,INX)               MSUB 44
      ENDIF                                                             MSUB 45
      IF(M.GT.1)THEN                                                    MSUB 46
           VY01=VY1(I,J,K)                                              MSUB 47
           VY02=VY2(I,J,K)                                              MSUB 48
           CALL   MSUB1(Y0,DYI,VY0,VY01,VY02,BY,TY,I,INY)               MSUB 49
      ENDIF                                                             MSUB 50
      IF(L.GT.1)THEN                                                    MSUB 51
           VZ01=VZ1(I,J,K)
```

Attachment 1, continued

```
      VZ02=VZ2(I,J,K)                                                    MSUB 52
      CALL   MSUB1(Z0,DZK,VZ0,VZ01,VZ02,BZ,TZ,K,INZ)                     MSUB 53
ENDIF                                                                    MSUB 54
TIME=AMIN1(TX,TY,TZ)                                                     MSUB 55
IF(TIME.EQ.1.E+38)THEN                                                   MSUB 56
      WRITE(1,3)I,J,K,IORG,JORG,KORG                                     MSUB 57
      WRITE(6,3)I,J,K,IORG,JORG,KORG                                     MSUB 58
      RETURN                                                             MSUB 59
ENDIF                                                                    MSUB 60
IF(TIME.LT.TX)INX=J                                                      MSUB 61
IF(TIME.LT.TY)INY=I                                                      MSUB 62
IF(TIME.LT.TZ)INZ=K                                                      MSUB 63
TISUM=TISUM+TIME                                                         MSUB 64
IF(N.GT.1)CALL  MSUB2(X,X0,DXJ,VX0,VX01,VX02,BX,TIME)                    MSUB 65
IF(M.GT.1)CALL  MSUB2(Y,Y0,DYI,VY0,VY01,VY02,BY,TIME)                    MSUB 66
IF(L.GT.1)CALL  MSUB2(Z,Z0,DZK,VZ0,VZ01,VZ02,BZ,TIME)                    MSUB 67
X0=X                                                                     MSUB 68
Y0=Y                                                                     MSUB 69
Z0=Z                                                                     MSUB 70
IF((INY.LT.1).OR.(INY.GT.M).OR.(INX.LT.1).OR.(INX.GT.N).OR.              MSUB 71
c (INZ.LT.1).OR.(INZ.GT.L).OR.(IH(INY,INX,INZ).NE.1))THEN                MSUB 72
      IDSYXZ=100*(100*INY+INX)+INZ                                       MSUB 73
      IF(LPATH)THEN                                                      MSUB 74
           WRITE(6,2)J,K,X,Z,TISUM                                       MSUB 75
           CALL PSUB(DISTX,DX,XSF,DISTZ,ZSF,DZ,                          MSUB 76
c          IPATH,PATH,NREC,NRLIM,KSYM,X,Z,J,K)                           MSUB 77
           TISUM=TISUM/365.25                                            MSUB 78
           WRITE(6,4)J,K,INX,INZ,TISUM                                   MSUB 79
      ENDIF                                                              MSUB 80
      RETURN                                                             MSUB 81
 ELSE                                                                    MSUB 82
      NMOVE=NMOVE+1                                                      MSUB 83
      IF(NMOVE.GT.LIMIT)THEN                                             MSUB 84
           WRITE(1,5)I,J,K,IORG,JORG,KORG                                MSUB 85
           WRITE(6,5)I,J,K,IORG,JORG,KORG                                MSUB 86
           RETURN                                                        MSUB 87
      ELSE                                                               MSUB 88
           IF(INX.GT.J)THEN                                              MSUB 89
                X0=0.                                                    MSUB 90
           ELSE IF(INX.LT.J)THEN                                         MSUB 91
                X0=DX(INX)                                               MSUB 92
           ENDIF                                                         MSUB 93
           IF(INY.GT.I)THEN                                              MSUB 94
                Y0=0.                                                    MSUB 95
           ELSE IF(INY.LT.I)THEN                                         MSUB 96
                Y0=DY(INY)                                               MSUB 97
           ENDIF                                                         MSUB 98
           IF(INZ.GT.K)THEN                                              MSUB 99
                Z0=0.                                                    MSUB100
           ELSE IF(INZ.LT.K)THEN                                         MSUB101
                Z0=DZ(INZ)                                               MSUB102
           ENDIF                                                         MSUB103
```

Attachment 1, continued

```
                J=INX                                                    MSUB104
                I=INY                                                    MSUB105
                K=INZ                                                    MSUB106
                IF(LPATH)THEN                                            MSUB107
                     WRITE(6,2)J,K,X0,Z0,TISUM                           MSUB108
                     CALL PSUB(DISTX,DX,XSF,DISTZ,ZSF,DZ,                MSUB109
     c                IPATH,PATH,NREC,NRLIM,KSYM,X0,Z0,J,K)              MSUB110
                ENDIF                                                    MSUB111
                GOTO1                                                    MSUB112
           ENDIF                                                         MSUB113
      ENDIF                                                              MSUB114
    2 FORMAT(1X,2I3,2F7.0,1P4E10.3)                                      MSUB115
    3 FORMAT(1H ,'STAGNATION POINT AT',3I5,' ORIGIN AT',3I5)             MSUB116
    4 FORMAT(1X,4I3,1PE10.3)                                             MSUB117
    5 FORMAT(1H ,'MOVE LIMIT REACHED AT ',3I5,' ORIGIN AT ',3I5)         MSUB118
      END                                                                MSUB119
```

Attachment 1, continued

```
C     MSUB1 - Determines position, velocity, and travel time to
C        next block face.
      SUBROUTINE MSUB1(W0,DW,V0,V1,V2,B,TIME,IW,INW)                    MSB1  1
      DV=V2-V1                                                          MSB1  2
      B=DV/DW                                                           MSB1  3
      IF((W0/DW).LT..5)THEN                                             MSB1  4
           V0=V1+B*W0                                                   MSB1  5
      ELSE                                                              MSB1  6
           V0=V2-B*(DW-W0)                                              MSB1  7
      ENDIF                                                             MSB1  8
      IF(V0.LT.0.)THEN                                                  MSB1  9
           IF(V1.LT.0.)THEN                                             MSB1 10
                INW=IW-1                                                MSB1 11
                VRL=ALOG(V1/V0)                                         MSB1 12
                IF(ABS(VRL).LT.1.5E-3)GOTO1                             MSB1 13
                TIME=VRL/B                                              MSB1 14
                IF(TIME.LE.0.)GOTO1                                     MSB1 15
           ELSE                                                         MSB1 16
                INW=IW                                                  MSB1 17
                TIME=1.E+38                                             MSB1 18
           ENDIF                                                        MSB1 19
      ELSE IF((V0.EQ.0.).OR.(V2.LE.0.))THEN                             MSB1 20
           INW=IW                                                       MSB1 21
           TIME=1.E+38                                                  MSB1 22
      ELSE                                                              MSB1 23
           INW=IW+1                                                     MSB1 24
           VRL=ALOG(V2/V0)                                              MSB1 25
           IF(ABS(VRL).LT.1.5E-3)GOTO1                                  MSB1 26
           TIME=VRL/B                                                   MSB1 27
           IF(TIME.LE.0.)GOTO1                                          MSB1 28
      ENDIF                                                             MSB1 29
      RETURN                                                            MSB1 30
    1 IF(V0.GT.0.)THEN                                                  MSB1 31
           INW=IW+1                                                     MSB1 32
           TIME=(DW-W0)/V0                                              MSB1 33
      ELSE                                                              MSB1 34
           INW=IW-1                                                     MSB1 35
           TIME=-W0/V0                                                  MSB1 36
      ENDIF                                                             MSB1 37
      RETURN                                                            MSB1 38
      END                                                               MSB1 39
```

Attachment 1, continued

```
C     MSUB2 - Determines new position after an increment of time.
      SUBROUTINE MSUB2(W,W0,DW,V0,V1,V2,B,TIME)                          MSB2  1
      IF(V0.EQ.0.)THEN                                                  MSB2  2
           W=W0                                                         MSB2  3
      ELSE                                                              MSB2  4
           BT=B*TIME                                                    MSB2  5
           IF(ABS(BT).LT.1.5E-3)THEN                                    MSB2  6
                WT=W0+V0*TIME                                           MSB2  7
           ELSE                                                         MSB2  8
                WT=(V0*EXP(BT)-V1)/B                                    MSB2  9
           ENDIF                                                        MSB2 10
           IF(WT.GT.DW)THEN                                             MSB2 11
                WT=DW                                                   MSB2 12
           ELSE IF(WT.LT.0.)THEN                                        MSB2 13
                WT=0.                                                   MSB2 14
           ENDIF                                                        MSB2 15
           IF(V1.EQ.0.)THEN                                             MSB2 16
                ZERO=0.                                                 MSB2 17
           ELSE IF(V2.EQ.0.)THEN                                        MSB2 18
                ZERO=DW                                                 MSB2 19
           ELSE                                                         MSB2 20
                IF(B.EQ.0.)THEN                                         MSB2 21
                     W=WT                                               MSB2 22
                     RETURN                                             MSB2 23
                ELSE                                                    MSB2 24
                     ZERO=-V1/B                                         MSB2 25
                ENDIF                                                   MSB2 26
           ENDIF                                                        MSB2 27
           IF(V0.LT.0.)THEN                                             MSB2 28
                IF((ZERO.GE.W0).OR.(ZERO.LT.0.))THEN                    MSB2 29
                     W=AMAX1(WT,0.)                                     MSB2 30
                ELSE                                                    MSB2 31
                     CALL ZER(WTT,ZERO,+1)                              MSB2 32
                     W=AMAX1(WT,WTT)                                    MSB2 33
                ENDIF                                                   MSB2 34
           ELSE IF((ZERO.LE.W0).OR.(ZERO.GT.DW))THEN                    MSB2 35
                W=AMIN1(WT,DW)                                          MSB2 36
           ELSE                                                         MSB2 37
                 CALL ZER(WTT,ZERO,-1)                                  MSB2 38
                 W=AMIN1(WT,WTT)                                        MSB2 39
           ENDIF                                                        MSB2 40
      ENDIF                                                             MSB2 41
      RETURN                                                            MSB2 42
      END                                                               MSB2 43
```

Attachment 1, continued

```
                IF(JM.GT.4194304)THEN                                   ZER 51
                    IF(JE.LT.255)THEN                                   ZER 52
                        JM=4194303                                      ZER 53
                        JE=JE+1                                         ZER 54
                    ELSE                                                ZER 55
                        JM=0                                            ZER 56
                    ENDIF                                               ZER 57
                ENDIF                                                   ZER 58
            ENDIF                                                       ZER 59
        ENDIF                                                           ZER 60
        JM=OR(JM,JS)                                                    ZER 61
        DO 2 ISUB=1,3                                                   ZER 62
      2 CR(ISUB)=CM(ISUB+1)                                             ZER 63
        CR(4)=CE(4)                                                     ZER 64
        WNEAR=R                                                         ZER 65
        RETURN                                                          ZER 66
        END                                                             ZER 67
```

Attachment 1, continued

```
C     ZER - Determines a position near, but upstream from, a point of
C        zero velocity.                                                    ZER  1
      SUBROUTINE ZER(WNEAR,WZERO,IDIR)                                    ZER  2
      CHARACTER CR(4),CM(4),CE(4),CS(4)                                   ZER  3
      EQUIVALENCE (R,CR(1)),(JM,CM(1)),(JE,CE(1)),(JS,CS(1))              ZER  4
      JM=0                                                                ZER  5
      JE=0                                                                ZER  6
      JS=0                                                                ZER  7
      R=WZERO                                                             ZER  8
      CE(4)=CR(4)                                                         ZER  9
      CS(2)=CR(1)                                                         ZER 10
      JS=AND(JS,8388608)                                                  ZER 11
      IF(R.EQ.0.)THEN                                                     ZER 12
           IF(IDIR.GT.0)THEN                                              ZER 13
                JM=4194304                                                ZER 14
           ELSE                                                           ZER 15
                JM=4194303                                                ZER 16
                JS=8388608                                                ZER 17
           ENDIF                                                          ZER 18
      ELSE                                                                ZER 19
           DO 1 ISUB=1,3                                                  ZER 20
    1      CM(ISUB+1)=CR(ISUB)                                            ZER 21
           JM=AND(JM,8388607)                                             ZER 22
           JM=JM+IDIR                                                     ZER 23
           IF(JS.EQ.0)THEN                                                ZER 24
                IF(JM.GT.8388607)THEN                                     ZER 25
                     IF(JE.LT.255)THEN                                    ZER 26
                          JM=4194304                                      ZER 27
                          JE=JE+1                                         ZER 28
                     ELSE                                                 ZER 29
                          JM=8388607                                      ZER 30
                     ENDIF                                                ZER 31
                ENDIF                                                     ZER 32
                IF(JM.LT.4194304)THEN                                     ZER 33
                     IF(JE.GT.0)THEN                                      ZER 34
                          JM=8388607                                      ZER 35
                          JE=JE-1                                         ZER 36
                     ELSE                                                 ZER 37
                          JM=0                                            ZER 38
                     ENDIF                                                ZER 39
                ENDIF                                                     ZER 40
           ELSE                                                           ZER 41
                JM=AND(JM,16777215)                                       ZER 42
                IF(JM.EQ.4194304)THEN                                     ZER 43
                     IF(JE.GT.0)THEN                                      ZER 44
                          JM=0                                            ZER 45
                          JE=JE-1                                         ZER 46
                     ELSE                                                 ZER 47
                          JM=0                                            ZER 48
                          JS=0                                            ZER 49
                     ENDIF                                                ZER 50
                ENDIF
```

Attachment 1, continued

```
C     PATH - Traces selected flow lines from recharge area to
C        discharge area.                                                    PATH   1
      SUBROUTINE PATH(RCH,IH,VX1,VX2,VZ1,VZ2,                             PATH   2
     c DX,DY,DZ,DISTX,DISTZ,RJ,NREC,PATH,Q,NP)                            PATH   3
      LOGICAL LFIN,LHCPT,LVPT,LLEFT,LVSW,LPATH                            PATH   4
      REAL KT                                                             PATH   5
      CHARACTER PATH(1)                                                   PATH   6
      CHARACTER*8 XMESUR,ZMESUR                                           PATH   7
      DIMENSION RCH(N),IH(N,L),VX1(N,L),VX2(N,L),VZ1(N,L),VZ2(N,L),       PATH   8
     c DX(N),DY(M),DZ(L),DISTX(N),DISTZ(L)                                PATH   9
      DIMENSION RJ(N,3),PER(7),Q(NEQ,6),NP(N,L)                           PATH  10
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,               PATH  11
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                  PATH  12
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR    PATH  13
      DATA LPATH/.TRUE./                                                  PATH  14
      DATA PER/.5,.1,.01,.001,.0001,.00001,.000001/                       PATH  15
      DATA NPER/7/                                                        PATH  16
      Y0=DY(1)/2.                                                         PATH  17
      IY=1                                                                PATH  18
      NRLIM=NREC                                                          PATH  19
      IPATH=0                                                             PATH  20
      NREC=0                                                              PATH  21
      LLEFT=.FALSE.                                                       PATH  22
      LVSW=.FALSE.                                                        PATH  23
      LFIN=.FALSE.                                                        PATH  24
      RL=0.                                                               PATH  25
      XSF=DINCHX*XSCALE                                                   PATH  26
      ZSF=DINCHZ*ZSCALE                                                   PATH  27
      RJS=0.                                                              PATH  28
      DO 2 J=1,N                                                          PATH  29
      DO 1 K=1,L                                                          PATH  30
      IF(IH(J,K).EQ.1)THEN                                                PATH  31
           IEQ=NP(J,K)                                                    PATH  32
           IF((J.GT.1).AND.(IH(J-1,K).EQ.2).AND.                          PATH  33
     c     (Q(IEQ,3).GT.0.))RJS=RJS+Q(IEQ,3)                              PATH  34
           RJ(J,1)=RJS                                                    PATH  35
           IF((K.GT.1).AND.(IH(J,K-1).EQ.2).AND.                          PATH  36
     c     (Q(IEQ,5).GT.0.))THEN                                          PATH  37
                RJS=RJS+Q(IEQ,5)                                          PATH  38
           ELSE                                                           PATH  39
                RJS=RJS+RCH(J)                                            PATH  40
           ENDIF                                                          PATH  41
           RJ(J,2)=RJS                                                    PATH  42
           IF((J.LT.N).AND.(IH(J+1,K).EQ.2).AND.                          PATH  43
     c     (Q(IEQ,4).GT.0.))RJS=RJS+Q(IEQ,4)                              PATH  44
           RJ(J,3)=RJS                                                    PATH  45
           GOTO2                                                          PATH  46
      ENDIF                                                               PATH  47
    1 CONTINUE                                                            PATH  48
    2 CONTINUE                                                            PATH  49
      WRITE(6,16)(J,(RJ(J,JJ),JJ=1,3),J=1,N)                              PATH  50
      DISTX(1)=0.
```

Attachment 1, continued

```
      DO 3 J=2,N                                                        PATH 51
    3 DISTX(J)=DISTX(J-1)+DX(J-1)                                       PATH 52
      DISTZ(L)=DZ(L)                                                    PATH 53
      DO 4 K=L-1,1,-1                                                   PATH 54
    4 DISTZ(K)=DISTZ(K+1)+DZ(K)                                         PATH 55
      DO 5 J=1,N                                                        PATH 56
      DO 5 K=1,L                                                        PATH 57
      IF(IH(J,K).EQ.1)GOTO6                                             PATH 58
    5 CONTINUE                                                          PATH 59
    6 IF((VZ1(J,K).LT.0.).OR.(VX1(J,K).LT.0.))LLEFT=.TRUE.              PATH 60
      J=1                                                               PATH 61
      K=0                                                               PATH 62
    7 IF(LFIN)RETURN                                                    PATH 63
      IF(J.EQ.1)THEN                                                    PATH 64
           K=K+1                                                        PATH 65
           IF(K.GT.L)THEN                                               PATH 66
                K=L                                                     PATH 67
                J=J+1                                                   PATH 68
                TEST=VX1(J,K)*VX2(J,K)                                  PATH 69
           ELSE                                                         PATH 70
                TEST=VZ1(J,K)*VZ2(J,K)                                  PATH 71
           ENDIF                                                        PATH 72
      ELSE IF(K.EQ.L)THEN                                               PATH 73
           J=J+1                                                        PATH 74
            IF(J.GT.N)THEN                                              PATH 75
                J=N                                                     PATH 76
                K=K-1                                                   PATH 77
                TEST=VZ1(J,K)*VZ2(J,K)                                  PATH 78
           ELSE                                                         PATH 79
                TEST=VX1(J,K)*VX2(J,K)                                  PATH 80
           ENDIF                                                        PATH 81
      ELSE IF(J.EQ.N)THEN                                               PATH 82
           K=K-1                                                        PATH 83
           IF((K.EQ.0).OR.(IH(J,K).NE.1))THEN                           PATH 84
                RR=RJ(N,3)                                              PATH 85
                LFIN=.TRUE.                                             PATH 86
                GOTO10                                                  PATH 87
           ELSE                                                         PATH 88
                TEST=VZ1(J,K)*VZ2(J,K)                                  PATH 89
           ENDIF                                                        PATH 90
      ENDIF                                                             PATH 91
      IF(TEST.GE.0.)GOTO7                                               PATH 92
      JBEG=J                                                            PATH 93
      KBEG=K                                                            PATH 94
      IF((J.GT.1).AND.(J.LT.N))THEN                                     PATH 95
           X0=-VX1(J,K)/(VX2(J,K)-VX1(J,K))*DX(J)                       PATH 96
           CALL ZER(Z0,DZ(L),-1)                                        PATH 97
      ELSE                                                              PATH 98
           Z0=-VZ1(J,K)/(VZ2(J,K)-VZ1(J,K))*DZ(K)                       PATH 99
           IF(J.EQ.1)THEN                                               PATH100
                CALL ZER(X0,0.,1)                                       PATH101
           ELSE IF(J.EQ.N)THEN                                          PATH102
```

Attachment 1, continued

```
                CALL ZER(X0,DX(N),-1)                                   PATH103
            ENDIF                                                       PATH104
        ENDIF                                                           PATH105
        IX=J                                                            PATH106
        IZ=K                                                            PATH107
        WRITE(6,17)J,K,LLEFT                                            PATH108
        IF(LLEFT)THEN                                                   PATH109
            LVSW=.TRUE.                                                 PATH110
            DO 8 J=1,N                                                  PATH111
            DO 8 K=1,L                                                  PATH112
            VX1(J,K)=-VX1(J,K)                                          PATH113
            VZ1(J,K)=-VZ1(J,K)                                          PATH114
            VX2(J,K)=-VX2(J,K)                                          PATH115
            VZ2(J,K)=-VZ2(J,K)                                          PATH116
    8       CONTINUE                                                    PATH117
        ENDIF                                                           PATH118
        KSYM=49                                                         PATH119
        CALL MSUB(X0,Y0,Z0,IY,IX,IZ,IDSYXZ,VX1,VX2,                     PATH120
       cVY1,VY2,VZ1,VZ2,DX,DY,DZ,IH,TISUM,LPATH,                        PATH121
       cDISTX,DISTZ,NREC,NRLIM,PATH,IPATH,KSYM,XSF,ZSF)                 PATH122
        IF(LVSW)THEN                                                    PATH123
            LVSW=.FALSE.                                                PATH124
            DO 9 J=1,N                                                  PATH125
            DO 9 K=1,L                                                  PATH126
            VX1(J,K)=-VX1(J,K)                                          PATH127
            VZ1(J,K)=-VZ1(J,K)                                          PATH128
            VX2(J,K)=-VX2(J,K)                                          PATH129
            VZ2(J,K)=-VZ2(J,K)                                          PATH130
    9       CONTINUE                                                    PATH131
        ENDIF                                                           PATH132
        J=IX                                                            PATH133
        K=IZ                                                            PATH134
        ZTEST=Z0/DZ(K)                                                  PATH135
        XTEST=X0/DX(J)                                                  PATH136
        IF(XTEST.LT..5)THEN                                             PATH137
            IF(XTEST.LT.ZTEST)THEN                                      PATH138
                RR=RJ(J-1,3)+ZTEST*                                     PATH139
       c        (RJ(J,1)-RJ(J-1,3))                                     PATH140
                WRITE(6,18)J,K,X0,Z0,RJ(J-1,3),RJ(J,1)                  PATH141
            ELSE                                                        PATH142
                RR=RJ(J,1)+XTEST*                                       PATH143
       c        (RJ(J,2)-RJ(J,1))                                       PATH144
                WRITE(6,18)J,K,X0,Z0,RJ(J,1),RJ(J,2)                    PATH145
            ENDIF                                                       PATH146
        ELSE                                                            PATH147
            IF((1.-XTEST).LT.ZTEST)THEN                                 PATH148
                RR=RJ(J,2)+ZTEST*                                       PATH149
       c        (RJ(J,3)-RJ(J,2))                                       PATH150
                WRITE(6,18)J,K,X0,Z0,RJ(J,2),RJ(J,3)                    PATH151
            ELSE                                                        PATH152
                RR=RJ(J,1)+XTEST*                                       PATH153
       c        (RJ(J,2)-RJ(J,1))                                       PATH154
```

Attachment 1, continued

```
                WRITE(6,18)J,K,X0,Z0,RJ(J,1),RJ(J,2)                    PATH155
            ENDIF                                                       PATH156
        ENDIF                                                           PATH157
   10 CONTINUE                                                          PATH158
      IF(LLEFT)THEN                                                     PATH159
           R1=RR                                                        PATH160
           R2=RL                                                        PATH161
      ELSE                                                              PATH162
           R1=RL                                                        PATH163
           R2=RR                                                        PATH164
      ENDIF                                                             PATH165
      WRITE(6,19)R1,R2                                                  PATH166
      KSYM=0                                                            PATH167
      DO 15 IPER=1,NPER                                                 PATH168
      RECH=R1+(R2-R1)*PER(IPER)                                         PATH169
      WRITE(6,20)RECH                                                   PATH170
      DO 11 J=1,N                                                       PATH171
      DO 11 JK=1,3                                                      PATH172
      IF(RJ(J,JK).GT.RECH)GOTO12                                        PATH173
   11 CONTINUE                                                          PATH174
   12 DO 13 K=1,L                                                       PATH175
      IF(IH(J,K).EQ.1)GOTO14                                            PATH176
   13 CONTINUE                                                          PATH177
   14 CONTINUE                                                          PATH178
      IF((J.EQ.1).AND.(JK.EQ.1))THEN                                    PATH179
           X0=DX(1)*RECH/RJ(1,1)                                        PATH180
           Z0=0.                                                        PATH181
      ELSE                                                              PATH182
           IF(JK.EQ.1)THEN                                              PATH183
                Z0=DZ(K)*(RECH-RJ(J-1,3))/(RJ(J,1)-RJ(J-1,3))           PATH184
                X0=0.                                                   PATH185
           ELSE IF(JK.EQ.2)THEN                                         PATH186
                X0=DX(J)*(RECH-RJ(J,1))/(RJ(J,2)-RJ(J,1))               PATH187
                Z0=0.                                                   PATH188
           ELSE                                                         PATH189
                Z0=DZ(K)*(RECH-RJ(J,2))/(RJ(J,3)-RJ(J,2))               PATH190
                X0=DX(J)                                                PATH191
           ENDIF                                                        PATH192
      ENDIF                                                             PATH193
      IF(X0.LT.0.)X0=0.                                                 PATH194
      IF(Z0.LT.0.)Z0=0.                                                 PATH195
      IF(X0.GT.DX(J))X0=DX(J)                                           PATH196
      IF(Z0.GT.DZ(K))Z0=DZ(K)                                           PATH197
      IX=J                                                              PATH198
      IZ=K                                                              PATH199
      CALL MSUB(X0,Y0,Z0,IY,IX,IZ,IDSYXZ,VX1,VX2,                       PATH200
     cVY1,VY2,VZ1,VZ2,DX,DY,DZ,IH,TISUM,LPATH,                          PATH201
     cDISTX,DISTZ,NREC,NRLIM,PATH,IPATH,KSYM,XSF,ZSF)                   PATH202
   15 CONTINUE                                                          PATH203
      RL=RR                                                             PATH204
      LLEFT=.NOT.LLEFT                                                  PATH205
      J=JBEG                                                            PATH206
```

Attachment 1, continued

```
      K=KBEG                                                          PATH207
      GOTO7                                                           PATH208
   16 FORMAT(1X,I2,1P3E11.3)                                          PATH209
   17 FORMAT(1X,2I3,L2)                                               PATH210
   18 FORMAT(1X,2I3,2F7.0,2F6.1)                                      PATH211
   19 FORMAT(' R1=',1PE10.3,',R2=',E10.3)                             PATH212
   20 FORMAT(' RECH=',1PE10.3)                                        PATH213
      END                                                             PATH214
```

Attachment 1, continued

```
C     PSUB - Saves coordinates along selected flow-paths.                    PSUB  1
      SUBROUTINE PSUB(DISTX,DX,XSF,DISTZ,ZSF,DZ,                             PSUB  2
     c IPATH,PATH,NREC,NRLIM,KSYM,X0,Z0,J,K)                                 PSUB  3
      DIMENSION DISTX(N),DX(N),DISTZ(L),DZ(L)                                PSUB  4
      CHARACTER PATH(1),IDXC(4),IDZC(4),SYM(50)                              PSUB  5
      LOGICAL LFIN,LHCPT,LVPT                                                PSUB  6
      REAL KT                                                                PSUB  7
      CHARACTER*8 XMESUR,ZMESUR                                              PSUB  8
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDM,MAX,L,M,N,LFIN,MN,                  PSUB  9
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                     PSUB 10
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR       PSUB 11
      EQUIVALENCE (IDX,IDXC(1)),(IDZ,IDZC(1))                                PSUB 12
      DATA SYM/'A','B','C','D','E','F','G','H','I','J','K','L','M','N',     PSUB 13
     c 'O','P','Q','R','S','T','U','V','W','X','Y','Z','a','b','c','d',     PSUB 14
     c 'e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t',     PSUB 15
     c 'u','v','w','x'/                                                      PSUB 16
      IF(NREC.EQ.NRLIM)THEN                                                  PSUB 17
           WRITE(1,1)                                                        PSUB 18
           WRITE(6,1)                                                        PSUB 19
           STOP                                                              PSUB 20
      ENDIF                                                                  PSUB 21
      IDX=6.*(DISTX(J)+X0)/XSF+1.5                                           PSUB 22
      IDZ=10.*(DISTZ(K)-Z0)/ZSF+3.5                                          PSUB 23
      IPATH=IPATH+1                                                          PSUB 24
      PATH(IPATH)=IDXC(3)                                                    PSUB 25
      IPATH=IPATH+1                                                          PSUB 26
      PATH(IPATH)=IDXC(4)                                                    PSUB 27
      IPATH=IPATH+1                                                          PSUB 28
      PATH(IPATH)=IDZC(4)                                                    PSUB 29
      IPATH=IPATH+1                                                          PSUB 30
      PATH(IPATH)=SYM(KSYM)                                                  PSUB 31
      NREC=NREC+1                                                            PSUB 32
      RETURN                                                                 PSUB 33
    1 FORMAT(' G8 ARRAY FILLED TO LIMIT OF 131072')                          PSUB 34
      END
```

Attachment 1, continued

```
C     SORT - Sorts flow-path coordinates.                                SORT  1
      SUBROUTINE SORT(IPOS,PATH,SRT,NREC,LREC)                           SORT  2
      CHARACTER PATH(1),A1,A2,SRT(1)                                     SORT  3
      DIMENSION IPOS(1)                                                  SORT  4
      JC=0                                                               SORT  5
      DO 1 IC=1,NREC                                                     SORT  6
      JC=JC+LREC                                                         SORT  7
      IPOS(IC)=JC                                                        SORT  8
    1 CONTINUE                                                           SORT  9
      IBEG=1                                                             SORT 10
      IEND=NREC                                                          SORT 11
    2 CONTINUE                                                           SORT 12
      ISW=0                                                              SORT 13
      ISW2=0                                                             SORT 14
      ISTART=IBEG+1                                                      SORT 15
      ILAST=IEND                                                         SORT 16
      IF(ILAST.LT.ISTART)GOTO7                                           SORT 17
      IP1=IPOS(IBEG)                                                     SORT 18
      J1=IP1-LREC                                                        SORT 19
      DO 4 IC=ISTART,ILAST                                               SORT 20
      IP2=IPOS(IC)                                                       SORT 21
      JJ1=J1                                                             SORT 22
      J2=IP2-LREC                                                        SORT 23
      JJ2=J2                                                             SORT 24
      DO 3 JC=1,LREC                                                     SORT 25
      JJ1=JJ1+1                                                          SORT 26
      JJ2=JJ2+1                                                          SORT 27
      A1=PATH(JJ1)                                                       SORT 28
      A2=PATH(JJ2)                                                       SORT 29
      IF(A1.GT.A2)THEN                                                   SORT 30
           ISW=IC                                                        SORT 31
           ISW2=1                                                        SORT 32
           IPOS(IC-1)=IP2                                                SORT 33
           GOTO4                                                         SORT 34
      ELSE IF(A1.LT.A2)THEN                                              SORT 35
           J1=J2                                                         SORT 36
           IF(ISW2.NE.0)THEN                                             SORT 37
                IPOS(IC-1)=IP1                                           SORT 38
                ISW2=0                                                   SORT 39
           ENDIF                                                         SORT 40
           IP1=IP2                                                       SORT 41
           GOTO4                                                         SORT 42
      ENDIF                                                              SORT 43
    3 CONTINUE                                                           SORT 44
    4 CONTINUE                                                           SORT 45
      IEND=IEND-1                                                        SORT 46
      IF(ISW.EQ.0)GOTO7                                                  SORT 47
      IF(ISW.GE.ILAST)THEN                                               SORT 48
           IPOS(ILAST)=IP1                                               SORT 49
      ENDIF                                                              SORT 50
      ISW=0                                                              SORT 51
      ISW2=0
```

Attachment 1, continued

```
      ISTART=IBEG                                                    SORT 52
      ILAST=IEND-1                                                   SORT 53
      IF(ILAST.LT.ISTART)GOTO7                                       SORT 54
      IP1=IPOS(IEND)                                                 SORT 55
      J1=IP1-LREC                                                    SORT 56
      DO 6 IB=ISTART,ILAST                                           SORT 57
      IC=ILAST-IB+ISTART                                             SORT 58
      IP2=IPOS(IC)                                                   SORT 59
      JJ1=J1                                                         SORT 60
      J2=IP2-LREC                                                    SORT 61
      JJ2=J2                                                         SORT 62
      DO 5 JC=1,LREC                                                 SORT 63
      JJ1=JJ1+1                                                      SORT 64
      JJ2=JJ2+1                                                      SORT 65
      A1=PATH(JJ1)                                                   SORT 66
      A2=PATH(JJ2)                                                   SORT 67
      IF(A1.LT.A2)THEN                                               SORT 68
           ISW=IC                                                    SORT 69
           ISW2=1                                                    SORT 70
           IPOS(IC+1)=IP2                                            SORT 71
           GOTO6                                                     SORT 72
      ELSE IF(A1.GT.A2)THEN                                          SORT 73
           J1=J2                                                     SORT 74
           IF(ISW2.NE.0)THEN                                         SORT 75
                IPOS(IC+1)=IP1                                       SORT 76
                ISW2=0                                               SORT 77
           ENDIF                                                     SORT 78
           IP1=IP2                                                   SORT 79
           GOTO6                                                     SORT 80
      ENDIF                                                          SORT 81
    5 CONTINUE                                                       SORT 82
    6 CONTINUE                                                       SORT 83
      IBEG=IBEG+1                                                    SORT 84
      IF(ISW.NE.0)THEN                                               SORT 85
           IF(ISW.GE.ISTART)THEN                                     SORT 86
                IPOS(ISTART)=IP1                                     SORT 87
           ENDIF                                                     SORT 88
           GOTO2                                                     SORT 89
      ENDIF                                                          SORT 90
    7 CONTINUE                                                       SORT 91
      ISRT=0                                                         SORT 92
      DO 9 IC=1,NREC                                                 SORT 93
      KC=IPOS(IC)-LREC+1                                             SORT 94
      KPLM1=K+LREC-1                                                 SORT 95
      DO 8 JC=KC,KPLM1                                               SORT 96
      ISRT=ISRT+1                                                    SORT 97
      SRT(ISRT)=PATH(JC)                                             SORT 98
    8 CONTINUE                                                       SORT 99
    9 CONTINUE                                                       SORT100
      RETURN                                                         SORT101
      END                                                            SORT102
```

Attachment 1, continued

```
C     PRINT - Constructs cross-sections showing selected flow-paths,
C        head, travel time, and lithology.
      SUBROUTINE PRINT(TIM,SRT,HD,IH,DX,DZ,SEC,NR)                          PRIN  1
      LOGICAL LFIN,LHCPT,LVPT                                               PRIN  2
      REAL KT                                                               PRIN  3
      REAL*8 HD(N,L),XN1                                                    PRIN  4
      CHARACTER*8 XLABEL,ZLABEL*35,XMESUR,ZMESUR                            PRIN  5
      CHARACTER*20 TITLE(4)                                                 PRIN  6
      CHARACTER VF4*24,VF3*24,VF1*21,VF5*22,VF2*21                          PRIN  7
      CHARACTER SRT(1),PRNT(123),SEC(N,L)                                   PRIN  8
      DIMENSION TIM(N,L),IH(N,L),DX(N),DZ(L)                                PRIN  9
      DIMENSION NA(4),XLABEL(3),ZN(13),XN(100),INDX(3)                      PRIN 10
      COMMON/VAR/NU,NL,IBW,NEQ,IGATE,MDH,MAX,L,M,N,LFIN,MN,                 PRIN 11
     c LHCPT,LVPT,IREF,SP,KT,TTOP,TEMPR,                                    PRIN 12
     c DELH,DELT,TCONV,HCONV,XSCALE,ZSCALE,DINCHX,DINCHZ,XMESUR,ZMESUR      PRIN 13
      DATA PRNT/123*' '/                                                    PRIN 14
      DATA XLABEL/' X DIS- ','TANCE IN',' MILES '/                          PRIN 15
      DATA ZLABEL/'DISTANCE IN Z DIRECTION IN MILES   '/                    PRIN 16
      DATA VF1/''(''(1H ,'',I3,''X,A20)'')'/                                PRIN 17
      DATA VF2/''(''(1H ,'',I3,''X,A35)'')'/                                PRIN 18
      DATA VF3/''(''(1H ,2X,A8,'',I3,''A1)'')'/                             PRIN 19
      DATA VF4/''(''(1H ,F10.2,'',I3,''A1)'')'/                             PRIN 20
      DATA VF5/''(''(1H ,10X,'',I3,''A1)'')'/                               PRIN 21
      DATA TITLE/' PLOT OF FLOWLINES  ','    PLOT OF HEAD                   PRIN 22
     c 'PLOT OF TRAVEL TIME ',' MODELED LITHOLOGY  '/                       PRIN 23
      DATA N1,N2,N3,NA(4),XN1/6,10,133,1000,.833333333D-1/                  PRIN 24
      DATA ISRT/0/                                                          PRIN 25
    1 XSF=DINCHX*XSCALE                                                     PRIN 26
      ZSF=DINCHZ*ZSCALE                                                     PRIN 27
      RXSF=XSF/N1                                                           PRIN 28
      RZSF=ZSF/N2                                                           PRIN 29
      XDIM=0.                                                               PRIN 30
      DO 2 J=1,N                                                            PRIN 31
    2 XDIM=XDIM+DX(J)                                                       PRIN 32
      ZDIM=0.                                                               PRIN 33
      DO 3 K=1,L                                                            PRIN 34
    3 ZDIM=ZDIM+DZ(K)                                                       PRIN 35
      NZD=ZDIM/ZSF                                                          PRIN 36
      IF(NZD*ZSF.LT.ZDIM)NZD=NZD+1                                          PRIN 37
      IF(NZD.GT.12)THEN                                                     PRIN 38
           NZD=12                                                           PRIN 39
           ZSF=ZDIM/NZD                                                     PRIN 40
           DINCHZ=ZSF/ZSCALE                                                PRIN 41
           RZSF=ZSF/N2                                                      PRIN 42
           WRITE(6,15)DINCHZ                                                PRIN 43
           WRITE(1,15)DINCHZ                                                PRIN 44
      ENDIF                                                                 PRIN 45
    4 NXD=XDIM/XSF+.5                                                       PRIN 46
      IF(NXD*XSF.LT.XDIM)NXD=NXD+1                                          PRIN 47
      NLIN=NXD*N1+1                                                         PRIN 48
      NEND=N1*XDIM/XSF+.5+1                                                 PRIN 49
      N5=NXD+1                                                              PRIN 50
```


READEX FICHE

Attachment 1, continued

```
      N6=NZD+1                                                          PRIN 51
      NPRNT=N2*ZDIM/ZSF+3.5                                             PRIN 52
      NA(1)=NLIN/2-1                                                    PRIN 53
      NA(2)=NLIN/2                                                      PRIN 54
      NA(3)=NLIN/2+3                                                    PRIN 55
      NB=NZD*5                                                          PRIN 56
      NC=NB-7                                                           PRIN 57
      IF(NC.LT.10)NC=10                                                 PRIN 58
      WRITE(VF1,VF1)NB                                                  PRIN 59
      WRITE(VF2,VF2)NC                                                  PRIN 60
      WRITE(VF3,VF3)NPRNT                                               PRIN 61
      WRITE(VF4,VF4)NPRNT                                               PRIN 62
      WRITE(VF5,VF5)NPRNT                                               PRIN 63
      XLABEL(3)=XMESUR                                                  PRIN 64
      ZLABEL(28:35)=ZMESUR                                              PRIN 65
      DO 5 I2=1,N6                                                      PRIN 66
    5 ZN(I2)=ZSF*(I2-1)/ZSCALE                                          PRIN 67
      DO 6 I2=1,N5                                                      PRIN 68
    6 XN(I2)=XSF*(I2-1)/XSCALE                                          PRIN 69
      DO 14 NG=1,4                                                      PRIN 70
      IF(NG.EQ.1)THEN                                                   PRIN 71
            NEXT=1                                                      PRIN 72
      ELSE IF(NG.EQ.2)THEN                                              PRIN 73
            CALL SCALEH(HD,N,L,FACT)                                    PRIN 74
            DO 7 J=1,N                                                  PRIN 75
            WRITE(6,17)(HD(J,K),K=L,1,-1)                               PRIN 76
    7       CONTINUE                                                    PRIN 77
            D1=DX(1)/2.                                                 PRIN 78
            NEXT=D1/RXSF+1.5                                            PRIN 79
      ELSE IF(NG.EQ.3)THEN                                              PRIN 80
            DO 8 J=1,N                                                  PRIN 81
            WRITE(6,18)(TIM(J,K),K=L,1,-1)                              PRIN 82
    8       CONTINUE                                                    PRIN 83
            CALL SCALET(TIM,N,L)                                        PRIN 84
            D1=DX(1)/2.                                                 PRIN 85
            NEXT=D1/RXSF+1.5                                            PRIN 86
      ELSE                                                              PRIN 87
            D2=DX(1)                                                    PRIN 88
            NEXT=1                                                      PRIN 89
      ENDIF                                                             PRIN 90
      J=1                                                               PRIN 91
      LL=1                                                              PRIN 92
      WRITE(6,VF1)TITLE(NG)                                             PRIN 93
      WRITE(6,VF2)ZLABEL                                                PRIN 94
      WRITE(6,19)(ZN(I2),I2=1,N6)                                       PRIN 95
      DO 13 ILIN=1,NLIN                                                 PRIN 96
      DO 9 IPRNT=1,NPRNT                                                PRIN 97
    9 PRNT(IPRNT)=' '                                                   PRIN 98
      IF(ILIN.GT.NEND)GOTO12                                            PRIN 99
      IF((ILIN.EQ.1).OR.(ILIN.EQ.NEND))THEN                             PRIN100
            DO 10 IPRNT=3,NPRNT                                         PRIN101
   10       PRNT(IPRNT)='-'                                            PRIN102
```

Attachment 1, continued

```
      DO 11 IPRNT=3,NPRNT,N2                                          PRIN103
11    PRNT(IPRNT)='+'                                                 PRIN104
      PRNT(NPRNT)='+'                                                 PRIN105
   ELSE                                                               PRIN106
      IF(MOD(ILIN,N1).EQ.1)THEN                                       PRIN107
         PRNT(3)='+'                                                  PRIN108
         PRNT(NPRNT)='+'                                              PRIN109
      ELSE                                                            PRIN110
         PRNT(3)='|'                                                  PRIN111
         PRNT(NPRNT)='|'                                              PRIN112
      ENDIF                                                           PRIN113
   ENDIF                                                              PRIN114
   IF(ILIN.NE.NEXT)GOTO12                                             PRIN115
   IF(NG.EQ.1)THEN                                                    PRIN116
      CALL PLINE1(ILIN,NEXT,ISRT,NR,PRNT,SRT,NPRNT)                   PRIN117
   ELSE IF(NG.EQ.2)THEN                                               PRIN118
      CALL PLINE2(ILIN,NEXT,HD,TIM,IH,PRNT,DX,DZ,D1,RXSF,RZSF,        PRIN119
c      NG,N,L,J)                                                      PRIN120
   ELSE IF(NG.EQ.3)THEN                                               PRIN121
      CALL PLINE2(ILIN,NEXT,HD,TIM,IH,PRNT,DX,DZ,D1,RXSF,RZSF,        PRIN122
c      NG,N,L,J)                                                      PRIN123
   ELSE                                                               PRIN124
      CALL PLINE4(ILIN,NEXT,SEC,IH,PRNT,DX,DZ,D2,RXSF,RZSF,N,L,J,     PRIN125
c     NPRNT)                                                          PRIN126
   ENDIF                                                              PRIN127
12 CONTINUE                                                           PRIN128
   IF(ILIN-NA(LL).EQ.0)THEN                                           PRIN129
      WRITE(6,VF3)XLABEL(LL),(PRNT(IPRNT),IPRNT=1,NPRNT)              PRIN130
      LL=LL+1                                                         PRIN131
   ELSE                                                               PRIN132
      IF(MOD(ILIN,N1).EQ.1)THEN                                       PRIN133
         INDEX=1+(ILIN-1)/6                                           PRIN134
         WRITE(6,VF4)XN(INDEX),(PRNT(IPRNT),IPRNT=1,NPRNT)            PRIN135
      ELSE                                                            PRIN136
         WRITE(6,VF5)(PRNT(IPRNT),IPRNT=1,NPRNT)                      PRIN137
      ENDIF                                                           PRIN138
   ENDIF                                                              PRIN139
13 CONTINUE                                                           PRIN140
   IF(NG.EQ.2)WRITE(6,20)FACT                                         PRIN141
   IF(NG.EQ.3)WRITE(6,21)                                             PRIN142
14 CONTINUE                                                           PRIN143
   RETURN                                                             PRIN144
15 FORMAT(' ',25X,10('*'),' TO FIT MAP WITHIN 12 INCHES, DINCHZ SHOULPRIN145
  cD BE GREATER THAN',G15.7,1X,10('*'))                               PRIN146
17 FORMAT(1X,20F6.1)                                                  PRIN147
18 FORMAT(1X,1P7E10.3)                                                PRIN148
19 FORMAT(1H ,5X,13F10.2)                                             PRIN149
20 FORMAT(' MULTIPLICATION FACTOR =',1PE8.1)                          PRIN150
21 FORMAT(' EXPONENT, 5 INDICATES 10.**5 <= VALUE < 10.**6')          PRIN151
   END                                                                PRIN152
```

Attachment 1, continued

```
C     PLINE1 - Constructs one line of flow-path cross-section.
      SUBROUTINE PLINE1(ILIN,NEXT,ISRT,NR,PRNT,SRT,NPRNT)                PLN1  1
      CHARACTER SRT(1),PRNT(1),BFX(4),BFZ(4)                             PLN1  2
      EQUIVALENCE (IBX,BFX(1)),(IBZ,BFZ(1))                              PLN1  3
      DATA IBX,IBZ/0,0/                                                  PLN1  4
    1 IF(ISRT.GE.NR)GOTO3                                                PLN1  5
      ISRT=ISRT+1                                                        PLN1  6
      BFX(3)=SRT(ISRT)                                                   PLN1  7
      ISRT=ISRT+1                                                        PLN1  8
      BFX(4)=SRT(ISRT)                                                   PLN1  9
      ISRT=ISRT+1                                                        PLN1 10
      BFZ(4)=SRT(ISRT)                                                   PLN1 11
      ISRT=ISRT+1                                                        PLN1 12
      IF(IBX.LT.ILIN)GOTO1                                               PLN1 13
      IF(IBX.GT.ILIN)GOTO2                                               PLN1 14
      IF((IBZ.LT.3).OR.(IBZ.GT.NPRNT))GOTO1                              PLN1 15
      PRNT(IBZ)=SRT(ISRT)                                                PLN1 16
      GOTO1                                                              PLN1 17
    2 ISRT=ISRT-4                                                        PLN1 18
      NEXT=IBX                                                           PLN1 19
      GOTO4                                                              PLN1 20
    3 NEXT=NPRNT+1                                                       PLN1 21
    4 RETURN                                                             PLN1 22
      END                                                                PLN1 23
```

Attachment 1, continued

```
C     SCALET - Scales travel time to three digits or less.
      SUBROUTINE SCALET(TIM,N,L)                                    SCLT  1
      DIMENSION TIM(N,L)                                            SCLT  2
      DO 2 K=1,L                                                    SCLT  3
      DO 2 J=1,N                                                    SCLT  4
      TIJK=TIM(J,K)                                                 SCLT  5
      IF(TIJK.LT.0.)GOTO3                                           SCLT  6
      IF(TIJK.GE.10.)GOTO1                                          SCLT  7
      ILOG=0                                                        SCLT  8
      GOTO2                                                         SCLT  9
    1 ILOG=ALOG10(TIJK)                                             SCLT 10
    2 TIM(J,K)=ILOG                                                 SCLT 11
    3 RETURN                                                        SCLT 12
      END                                                           SCLT 13
```

Attachment 1, continued

```
C     PLINE2 - Constructs one line of head or travel-time cross-section.
      SUBROUTINE PLINE2(ILIN,NEXT,HD,TIM,IH,PRNT,DX,DZ,D1,RXSF,RZSF,     PLN2  1
     c NG,N,L,J)                                                         PLN2  2
      CHARACTER PRNT(1),SYM(0:11),PRCH                                   PLN2  3
      REAL*8 HD(N,L)                                                     PLN2  4
      DIMENSION TIM(N,L),IH(N,L),DX(1),DZ(1),INDX(3)                     PLN2  5
      DATA SYM/'0','1','2','3','4','5','6','7','8','9','-',' '/         PLN2  6
      DISTZ=0.                                                           PLN2  7
      DO 4 K=L,1,-1                                                      PLN2  8
      IF(K.EQ.L)THEN                                                     PLN2  9
           DISTZ=DISTZ+DZ(L)/2.                                          PLN2 10
      ELSE                                                               PLN2 11
           DISTZ=DISTZ+(DZ(K+1)+DZ(K))/2.                                PLN2 12
      ENDIF                                                              PLN2 13
      IZ=DISTZ/RZSF+3.5                                                  PLN2 14
      IF(IH(J,K).NE.1)GOTO4                                              PLN2 15
      IF(NG.EQ.2)THEN                                                    PLN2 16
           VAR=HD(J,K)                                                   PLN2 17
      ELSE                                                               PLN2 18
           VAR=TIM(J,K)                                                  PLN2 19
      ENDIF                                                              PLN2 20
      IF(VAR.LT.0)THEN                                                   PLN2 21
           IN=-VAR+.5                                                    PLN2 22
           IF(IN.GE.10)THEN                                              PLN2 23
                INDX(3)=10                                               PLN2 24
                INDX(2)=IN/10                                            PLN2 25
                INDX(1)=MOD(IN,10)                                       PLN2 26
           ELSE                                                          PLN2 27
                INDX(3)=11                                               PLN2 28
                INDX(2)=10                                               PLN2 29
                INDX(1)=IN                                               PLN2 30
           ENDIF                                                         PLN2 31
      ELSE IF(VAR.EQ.0)THEN                                              PLN2 32
           INDX(1)=0                                                     PLN2 33
           INDX(2)=11                                                    PLN2 34
           INDX(3)=11                                                    PLN2 35
      ELSE                                                               PLN2 36
           IN=VAR+.5                                                     PLN2 37
           INDX(1)=MOD(IN,10)                                            PLN2 38
           IF(IN.GE.100)THEN                                             PLN2 39
                INDX(3)=IN/100                                           PLN2 40
                INDX(2)=MOD(IN,100)/10                                   PLN2 41
           ELSE IF(IN.GE.10)THEN                                         PLN2 42
                INDX(3)=11                                               PLN2 43
                INDX(2)=IN/10                                            PLN2 44
           ELSE                                                          PLN2 45
                INDX(3)=11                                               PLN2 46
                INDX(2)=11                                               PLN2 47
           ENDIF                                                         PLN2 48
      ENDIF                                                              PLN2 49
      IF(INDX(3).NE.11)THEN                                              PLN2 50
           I2=3                                                          PLN2 51
```

Attachment 1, continued

```
      ELSE IF(INDX(2).NE.11)THEN                                    PLN2 52
           I2=2                                                     PLN2 53
      ELSE                                                          PLN2 54
           I2=1                                                     PLN2 55
      ENDIF                                                         PLN2 56
      IF((IZ-I2).LT.0)GOTO4                                         PLN2 57
      DO 2 I1=1,I2                                                  PLN2 58
      PRCH=PRNT(IZ-I2+I1)                                           PLN2 59
      IF(PRCH.GE.'0'.AND.PRCH.LE.'9')GOTO4                          PLN2 60
    2 CONTINUE                                                      PLN2 61
      DO 3 I1=1,I2                                                  PLN2 62
    3 PRNT(IZ-I1+1)=SYM(INDX(I1))                                   PLN2 63
    4 CONTINUE                                                      PLN2 64
   45 J=J+1                                                         PLN2 65
      IF(J.LE.N)THEN                                                PLN2 66
           D1=D1+(DX(J-1)+DX(J))/2.                                 PLN2 67
           NEXT=D1/RXSF+1.5                                         PLN2 68
      ELSE                                                          PLN2 69
           NEXT=NZ+1                                                PLN2 70
           GOTO5                                                    PLN2 71
      ENDIF                                                         PLN2 72
      IF(NEXT.EQ.ILIN)GOTO45                                        PLN2 73
    5 RETURN                                                        PLN2 74
      END                                                           PLN2 75
```

Attachment 1, continued

```
C     SCALEH - Scales head to three digits or less.
      SUBROUTINE SCALEH(HD,N,L,FACT)                                    SCLH  1
      REAL*8 HD(N,L)                                                    SCLH  2
      HMAX=-1.E+35                                                      SCLH  3
      HMIN=1.E+35                                                       SCLH  4
      DO 1 K=1,L                                                        SCLH  5
      DO 1 J=1,N                                                        SCLH  6
      IF(HD(J,K).GT.HMAX)THEN                                           SCLH  7
           HMAX=HD(J,K)                                                 SCLH  8
      ELSE IF(HD(J,K).LT.HMIN)THEN                                      SCLH  9
           HMIN=HD(J,K)                                                 SCLH 10
      ENDIF                                                             SCLH 11
    1 CONTINUE                                                          SCLH 12
      IF(HMAX.LE.-HMIN)HMAX=-HMIN                                       SCLH 13
      IF(HMAX.EQ.0.)THEN                                                SCLH 14
           FACT=1.                                                      SCLH 15
      ELSE                                                              SCLH 16
           HLOG=ALOG10(HMAX)-2.                                         SCLH 17
           ILOG=HLOG                                                    SCLH 18
           IF(HLOG.LT.0.)ILOG=ILOG-1                                    SCLH 19
           IF(HMAX.EQ.-HMIN)ILOG=ILOG+1                                 SCLH 20
           FACT=10.**ILOG                                               SCLH 21
           DO 2 K=1,L                                                   SCLH 22
           DO 2 J=1,N                                                   SCLH 23
           HD(J,K)=HD(J,K)/FACT                                         SCLH 24
    2      CONTINUE                                                     SCLH 25
      ENDIF                                                             SCLH 26
      RETURN                                                            SCLH 27
      END                                                               SCLH 28
```

Attachment 1, continued

```
C     PLINE4 - Constructs one line of lithologic cross-section.
      SUBROUTINE PLINE4(ILIN,NEXT,SEC,IH,PRNT,DX,DZ,D2,RXSF,RZSF,N,L,J,  PLN4  1
                                                                         PLN4  2
     c NPRNT)                                                            PLN4  3
      CHARACTER SEC(N,L),PRNT(1)                                         PLN4  4
      DIMENSION IH(N,L),DX(1),DZ(1)                                      PLN4  5
      DISTX=(ILIN-1)*RXSF                                                PLN4  6
      IF(DISTX.GT.D2)THEN                                                PLN4  7
           J=J+1                                                         PLN4  8
           IF(J.LE.N)THEN                                                PLN4  9
                D2=D2+DX(J)                                              PLN4 10
           ELSE                                                          PLN4 11
                J=N                                                      PLN4 12
           ENDIF                                                         PLN4 13
      ENDIF                                                              PLN4 14
      K=L                                                                PLN4 15
      D3=0.                                                              PLN4 16
      D4=DZ(L)                                                           PLN4 17
      DO 2 IPRNT=3,NPRNT                                                 PLN4 18
      DISTZ=(IPRNT-3)*RZSF                                               PLN4 19
    1 IF(DISTZ.GT.D4)THEN                                                PLN4 20
           K=K-1                                                         PLN4 21
           IF(K.GT.0)THEN                                                PLN4 22
                D3=D4                                                    PLN4 23
                D4=D4+DZ(K)                                              PLN4 24
                GOTO1                                                    PLN4 25
           ELSE                                                          PLN4 26
                K=1                                                      PLN4 27
           ENDIF                                                         PLN4 28
      ENDIF                                                              PLN4 29
      IF(IH(J,K).NE.3)PRNT(IPRNT)=SEC(J,K)                               PLN4 30
    2 CONTINUE                                                           PLN4 31
      NEXT=ILIN+1                                                        PLN4 32
      RETURN                                                             PLN4 33
      END                                                                PLN4 34
```

Attachment 2. Definition of selected program variables

Attachment 2.

A - Array (NL*IBW) containing the reduced lower-right partition of the coefficient matrix.
AL - Array (6) containing the coefficients, for one block, in the lower-left partition of the matrix.
ARRAY - Input array to be read.
AU - Array (6*NU) containing the upper-right partition of the coefficient matrix.
B - The second parameter in the linear velocity function, v=a+b*w.
DELH - Largest change in head at last solution.
DELT - Largest change in temperature at last solution.
DELV - Largest change (either head or temperature) at last solution.
DINCHX - Number of horizontal scale units per inch, for output.
DINCHZ - Number of vertical scale units per inch, for output.
DISTX - Array (N) containing horizontal distance (ft) of the decreasing face of each block from the left edge of the section.
DISTZ - Array (N) containing vertical distance (ft) of the decreasing face of each block from the bottom of the section.
DP - Array (M*N) containing depths below surface for the top nodes of the model.
DU - Array (NU) containing the upper-left partition of the coefficient matrix.
DW - Distance across a model block, in the direction of one of the coordinate axes.
DX - Array (N) containing the dimensions of the blocks in the horizontal-x direction.
DY - Array (M) containing the dimensions of the blocks in the horizontal-y direction.
DZ - Array (L) containing the dimensions of the blocks in the vertical direction.
D1 - The distance, from the left side of the section, to the next line to be plotted.
D2 - The distance, from the left side of the section, to the next block face.
FACT - Scaling factor for the head values displayed on a cross-section.
G1 - Array containing HD (head), TMP (temperature), DX (horizontal-x dimensions), DY (horizontal-y dimensions), DZ (vertical dimensions), DISTX (horizontal distances), and DISTZ (vertical distances.
G2 - Array containing IH (block types for water equation), IT (block types for heat equation), NP (block number), IR (row number corresponding to the block number), JR (column number), and KR (layer number).
G3 - Array containing Q (interblock ground- water flow).
G4 - Array containing HC (hydraulic conductivity at computed temperature), P (hydraulic conductivity at reference temperature), POR (porosity), RCH (recharge from rainfall), RJ (recharge from all sources at the top of the model), DP (depth below surface for the top model layer), and HF (geothermal heat flow at the bottom of the model).
G5 - Array containing AU (upper-right partition of the coefficient matrix).

Attachment 2, continued

G6 - Array containing DU (upper-left partition of the coefficient matrix), RU (upper part of the known vector), and RL (lower part of known vector).
G7 - Array containing VX1 and VX2 (x velocities), VY1 and VY2 (y velocities), and VZ1 and VZ2 (z velocities).
G8 - Array containing SEC (block lithology), TIM (travel time), IDS (disharge points), PATH (flow path coordinates), IPOS (data for sorting routine), and SRT (sorted coordinates).
HD - Array (M*N*L) containing elevation of the water level for a node.
HC - Array (M*N*L) containing hydraulic conductivity adjusted to the temperature at each block.
HCONV - Convergence test for head.
solution if changes in computed head are smaller.
HF - Array (M*N) containing heat-flow rates at a node in the bottom layer of the model.
I - Location index in the y direction.
IAU - Pointer to the AU array.
IBW - Bandwidth of reduced lower-right part of coefficient matrix.
IDIR - The upstream direction near a stagnation point, +1 indicates increasing direction, -1 indicates decreasing direction.
IDISTX - Pointer to the DISTX array.
IDISTZ - Pointer to the DISTZ array.
IDP - Pointer to the DP array.
IDS - Array (M*N*L) containing discharge points for each node.
IDSYXZ - The discharge point for one beginning point.
IDU - Pointer to the DU array.
IDX - Pointer to the DX array
IDY - Pointer to the DY array
IDZ - Pointer to the DZ array.
IEQ - Index indicating equation number.
IGATE - Number of times equations have been solved.
IH - Array (M*N*L) containing integer codes indicating the type of model block for the ground-water flow equation. Codes are: 1, active block, 2, constant-head, and 3, inactive block.
IHC - Pointer to the HC array.
IHD - Pointer to the HD array.
IHF - Pointer to the HF array.
IIDS - Pointer to the IDS array.
IIH - Pointer to the IH array.
IIR - Pointer to the IR array.
IIT - Pointer to the IT array.
IJR - Pointer to the JR array.
IKR - Pointer to the KR array.
IL - Index indicating equation number relative to lower partition.
ILIN - Number (horizontal direction) of a line on a cross-section.
INP - Pointer to the NP array.
INW - New block number (in one direction) after a move, either one greater or less than previous number.
IQ - Pointer to the Q array.
IP - Pointer to the P array.
IPATH - Pointer to location within PATH array.

Attachment 2, continued

IPOR - Pointer to the POR array.
IPOS - Array (NREC) containing pointers to flow-path data.
IR - Array (NEQ) containing pointers from equation number to row.
IRCH - Pointer to the RCH array.
IREF - Reference temperature (Celsius) for input values of
hydraulic conductivity.
IRJ - Pointer to the RJ array.
IRL - Pointer to the RL array.
IRU - Pointer to the RU array.
ISEC - Pointer to the SEC array.
ISRT - Pointer to location within SRT array.
IT - Array (M*N*L) containing integer codes indicating the type
of model block for the heat flow equation. Codes are: 1, active
block, 2, constant temperature, and 3, inactive block.
ITIM - Pointer to the TIM array.
ITMP - Pointer to the TMP array.
IVX1 - Pointer to the VX1 array.
IVX2 - Pointer to the VX2 array.
IVY1 - Pointer to the VY1 array.
IVY2 - Pointer to the VY2 array.
IVZ1 - Pointer to the VZ1 array.
IVZ2 - Pointer to the VZ2 array.
IW - Location index (in one direction) before a move.
J - Location index in the x direction.
JR - Array (NEQ) containing pointers from equation number to column.
K - Location index in the z direction.
KL - Number of layers for an array, KL=1 for DP, RCH, and HF, and KL=L
for other arrays.
KR - Array (NEQ) containing pointers from equation number to layer.
KSYM - Number for flow-path symbol.
KT - Thermal conductivity, input in conductance units
(10**-3 cal./(cm.*s*deg. C).
L - Number of blocks in the vertical direction.
LFIN - Logical, T indicates final solution.
LHCPT - Logical, LHCPT=T displays values of adjusted hydraulic
conductivity in "input-file-name".BAL.
LMN - L*M*N, number of blocks in the model.
LPATH - Logical, LPATH=T indicates that flow paths are to be traced.
LREC - Number of bytes in one flow-path coordinate.
LSTOP - Logical, T causes program stop due to insufficient
storage allocation.
LVPT - Logical, LVPT=T displays velocities in "input-file-name".VELO.
M - Number of blocks in the horizontal-y direction.
MAX - Maximum repetitions of solution procedure.
MD - Row position of the main diagonal term in the A matrix.
MDM - IBW/2 (half bandwidth).
MN - M*N (number of rows * number of columns).
N - Number of blocks in the horizontal-x direction.
NEQ - Number of active blocks (equations).
NEXT - Next line of the output section that will display data.
NG - Section counter in PRINT routine, 1 - flow paths, 2 - head,
3 - travel time, and 4 - lithology.

NL - Number of blocks (equations) in the lower partition of the matrix.
NP - Array (M*N*L) containing pointers from row, column, and layer to equation number.
NR - Number of bytes in PATH or SRT arrays, equal to 4*NREC.
NREC - Number of flow-path coordinates.
NRLIM - Number of bytes available in G8 array.
NU - Number of blocks (equations) in the upper partition of the matrix.
NPRNT - Number of characters in one vertical line of the output sections.
P - Array (M*N*L) containing hydraulic conductivity for each block at the reference temperature.
PATH - Array (NR bytes) containing flow-path data.
POR - Array (M*N*L) containing porosity for each block.
PRNT - Characters in one vertical line of a cross-section.
Q - Array (NEQ*6) containing discharge of water for each active block through each of the block faces.
RCH - Array (M*N) containing recharge rate for the top nodes.
RJ - Array (N*3) containing inflow to the top of a section from recharge and constant-head nodes.
RL - Array (NL) containing the lower part of the known vector.
RU - Array (NU) containing the upper part of the known vector.
RXSF - Horizontal distance represented by one character on a cross-section.
RZSF - Vertical distance represented by one character on a cross-section.
SEC - Array (N*L bytes) containing one-character codes representing the lithologic type of each block.
SP - Volumetric specific heat for fluid, input in btu/(ft**3*deg.f)
SRT - Array (NR bytes) containing sorted flow-path data.
TMP - Array (M*N*L) containing temperatures for a node.
TCONV - Convergence test for temperature.
solution if changes in computed temperature are smaller.
TEMPR - Temperature of recharge from rainfall, deg. F.
TIM -Array (M*N*L) containing time of travel (years) from a node to a discharge point.
TIME - The time of travel (days) across a block.
TISUM - The time of travel (days) along a flow path, equal to the sum of TIME along the path.
TTOP - Surface temperature, deg. f.
V - Either head or temperature, depending on which equation is solved.
VAR - Common block containing several variables needed in subroutines.
VX1 - Array (M*N*L) containing x velocity component at the face between i,j-1,k and i,j,k. Positive if movement is from i,j-1,k to i,j,k.
VX2 - Array (M*N*L) containing x velocity component at the face between i,j,k and i,j+1,k. Positive if movement is from i,j,k to i,j+1,k.
VY1 - Array (M*N*L) containing y velocity component at the face between i-1,j,k and i,j,k. Positive if movement is from i-1,j,k to i,j,k.
VY2 - Array (M*N*L) containing y velocity component at the face between i,j,k and i+1,j,k. Positive if movement is from i,j,k to i+1,j,k.
VZ1 - Array (M*N*L) containing z velocity component at the face between i,j,k-1 and i,j,k. Positive if movement is from i,j,k-1 to i,j,k.

Attachment 2, continued

VZ2 - Array (M*N*L) containing z velocity component at the face between i,j,k and i,j,k+1. Positive if movement is from i,j,k to i,j,k+1.
V0 - Velocity component, at a location (W0) within a block.
V1 - Velocity component, at a decreasing block face.
V2 - Velocity component, at an increasing block face.
W - Distance from decreasing block-face after a move step. Value is from zero up to the dimension of the block.
WNEAR - A location near a stagnation point but upstream from it.
WZERO - A stagnation point (point of zero velocity).
W0 - Distance from decreasing block-face before a move step. Represents X0, Y0, or Z0. Value is from zero to the dimension of the block.
XMESUR - Name of the horizontal scale unit, for output cross sections.
XSCALE - Number of feet in the horizontal scale unit, for output.
XSF - Horizontal scale (ft/in.) for the cross-sections.
X0 - Distance (ft) from the decreasing block-face in the x direction.
Y0 - Distance (ft) from the decreasing block-face in the y direction.
ZMESUR - Name of the vertical scale unit, for output cross sections.
ZSCALE - Number of feet in the vertical scale unit, for output.
ZSF - Vertical scale (ft/in.) for the cross-sections.
Z0 - distance (ft) from the decreasing block-face in the z direction.

Attachment 3. Input data

Attachment 3.

The input data was designed for card format (80 character lines) although the input data is read from disk files. For number of lines, the ratio A/B, if not an integer, is rounded up to the next larger integer. Explanation of the formats are in FORTRAN language manuals. HOTWTR prompts for the input file name and reads the data from it. The input file name must not be more than 12 characters.

Number of lines	Columns	Variable	Format	Definition
1	1-3	M	I3	Number of blocks in the y direction
	4-6	N	I3	Number of blocks in the x direction
	7-9	L	I3	Number of blocks in the z direction
	10-12	MAX	I3	Number of iterations
	13-15	IREF	I3	Base temperature, deg. C, for hydraulic conductivity
	16	LHCPT	L1	Display option for hydraulic conductivity, T=yes, F=no
	17	LVPT	L1	Display option for velocities, T=yes, F=no

Note.-None of M, N, or L should be greater than 99. M = 1 produces special output for cross-sections. MAX = 1 indicates a solution for the ground-water equation only. For a solution to both equations, MAX = 5 results in adequate solutions for many cross-section models. IREF indicates the temperature for the input hydraulic conductivity. Changes in temperature from IREF modify the hydraulic conductivity by changes in the kinematic viscosity of water. Viscosity data used in the program is from Lohman and others (1972, table 3).

Attachment 3, continued

Number of lines	Columns	Variable	Format	Definition
1	1-5	KT	F5.0	Thermal conductivity, in conductance units (0.001 calories/(cm.*sec.)
	6-10	SP	F5.0	Volumetric specific heat, BTU/cubic ft.
	11-15	TTOP	F5.0	Temperature, deg. F, at the point of conductive heat discharge out of the top of the model.
	16-20	TEMPR	F5.0	Temperature, deg. F, of recharge from precipitation.
	21-25	TCONV	F5.0	Temperature-change criterion, deg. F
	26-30	HCONV	F5.0	Head-change criterion, ft.
N/16	1-80	DX(J)	16F5.0	Block lengths, ft, x direction.
M/16	1-80	DY(I)	16F5.0	Block lengths, ft, y direction.
L/16	1-80	DZ(K)	16F5.0	Block lengths, ft, z direction.

Attachment 3, continued

Number of lines	Columns	Variable	Format	Definition
L*M*(N/80)	1-80	IT(I,J,K)	80I1	Block type for the heat-flow model.

Note.-The types are: 1, active node (unknown temperature to be solved for), 2, constant temperature node, and 3, no heat conduction across the block face but there may be convection across the block face. The data are grouped by layer (z index or K), by row (y index or I) within each layer, and by column (x index or J) within each row. Each row begins on a new line.

L*M*(N/80)	1-80	IH(I,J,K)	80I1	Block type for the ground-water-flow model.

Note.-The types are: 1, active node, 2, constant head, and 3, no flow across block face. Active nodes should be the same for heat and water ($IT_{i,j,k}$=1 requires that $IH_{i,j,k}$=1). However, inactive nodes do not have to correspond ($IH_{i,j,k}$=2 but $IT_{i,j,k}$=3, for example). At least one node in the ground-water model must be constant head because head cannot be determined by flux alone. No constant-temperature nodes are necessary. The sequence is the same as for IT.

Attachment 3, continued

Each of the following data sets consists of a parameter card and, if the data set contains variable data, additional data cards. Each parameter card is coded as:

Number of lines	Columns	Variable	Format	Definition
1	1-10	CONS	F10.0	If IVAR=0 then all array values are equal to CONS. Otherwise, CONS is a multiplication factor for the data values that follow.
	11-15	IVAR	I5	If IVAR=0 then no data cards follow. Otherwise, data cards for this array follow.

Attachment 3, continued

When data cards are included, start each row on a new card. Multi-layered data sets are coded by layers. The first three of the following data sets are always one layer.

Number of lines	Columns	Variable	Format	Definition
M*(N/20)	1-80	DP(I,J)	20F4.0	Distance, ft, along the path of conductive heat-discharge out the top of the model.

Note.-If TTOP is the average air temperature then DP is the distance that the top active node is below land surface.

M*(N/20)	1-80	RCH(I,J)	20F4.0	Recharge, in./yr, from precipitation.

Note.-Recharge should be coded as zero if the top active node is overlain by a constant-head node.

M*(N/20)	1-80	HF(I,J)	20F4.0	Geothermal heat-flow, heat flow units (0.000001 calories /(square cm*sec)).
L*M*(N/20)	1-80	HD(I,J,K)	20F4.0	Head, ft.

Note.-Only head at constant-head nodes need be coded. Other heads can be blank. If there is only one constant-head node, it's head can be coded on the parameter card and the data cards can be omitted.

L*M*(N/20)	1-80	TMP(I,J,K)	20F4.0	Temperature, deg. F.

Note.-Values should be specified for constant-temperature nodes (IT=2) or for constant-head nodes (IH=2) where flow is into the model.

Attachment 4. Description of output data files.

Attachment 4.

OUTPUT FILE NAME[1]	DESCRIPTION OF FILE
"FILE".INPUT	Contains programs interpretation of all input data; equation, row, column, and layer numbers.
"FILE".BAL	Heat and water balance, calculated temperature and head, and hydraulic conductivity (if LHCPT=T in input data).
"FILE".VELO	Velocities (if LVPT=T in input data) for x, y, and z directions.
"FILE".MOVE	Time of travel to discharge node, and location of discharge node.
"FILE".PATH	Inflow at top of model section, flow divides (if any), and points on selected flow paths.
"FILE".PRINT	Line-printer drawn sections showing selected flow paths, head, time of travel, and lithology.

[1] "FILE" will be replaced by the name of the input data.

Attachment 5. Example simulation

Attachment 5.

This simulation is for the cross-section shown on the last page of this attachment. The section is 200,000 ft long and 7,500 ft in vertical extent. Three rock types are shown on the section: basalt (B), crystalline rocks (G), and alluvial fill (A). There are two discharge areas for ground water, one at each end of the section. The basalt has a hydraulic conductivity of .5 ft/d and a porosity of .05. The crystalline rocks have a hydraulic conductivity of .0001 ft/d and porosity of .0001. The alluvial fill has a hydraulic conductivity of 10 ft/d and porosity of .3. The water-level elevation in both of the discharge areas is 1000 ft. The section is divided into 20 model blocks in the horizontal direction and 10 blocks in the vertical. The input data for this simulation is as follows.

Attachment 5, continued

```
    ---------------------------------------------------------------------------------------------
    |  1 20 10  5 20FF
    |   6. 62.5  68.  68.  .01  .01
    ---------------------------------------------------------------------------------------------
DX  |  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4  1+4
    |  1+4  1+4  1+4  1+4
    ---------------------------------------------------------------------------------------------
DY  |1.
    ---------------------------------------------------------------------------------------------
DZ  |  500  500  500  500  500  500  500 1000 1000 2000
    ---------------------------------------------------------------------------------------------
    |33333333111133333333
    |33333331111113333333
    |33333311111111333333
    |11111111111111111111
IT  |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    ---------------------------------------------------------------------------------------------
    |33333333111133333333
    |33333331111113333333
    |23333311111111333332
    |11111111111111111111
IH  |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    |11111111111111111111
    ---------------------------------------------------------------------------------------------
DP  |250.
    ---------------------------------------------------------------------------------------------
RCH|.05       1
    |   0   1   1   1   1   1 1-3 2-1 2-1 2-1 2-1 2-1 2-1 1-3   1   1   1   1   1   0
    ---------------------------------------------------------------------------------------------
HF  |1.5
    ---------------------------------------------------------------------------------------------
HD  |1000.
    ---------------------------------------------------------------------------------------------
TMP|68.
    ---------------------------------------------------------------------------------------------
```

Attachment 5, continued

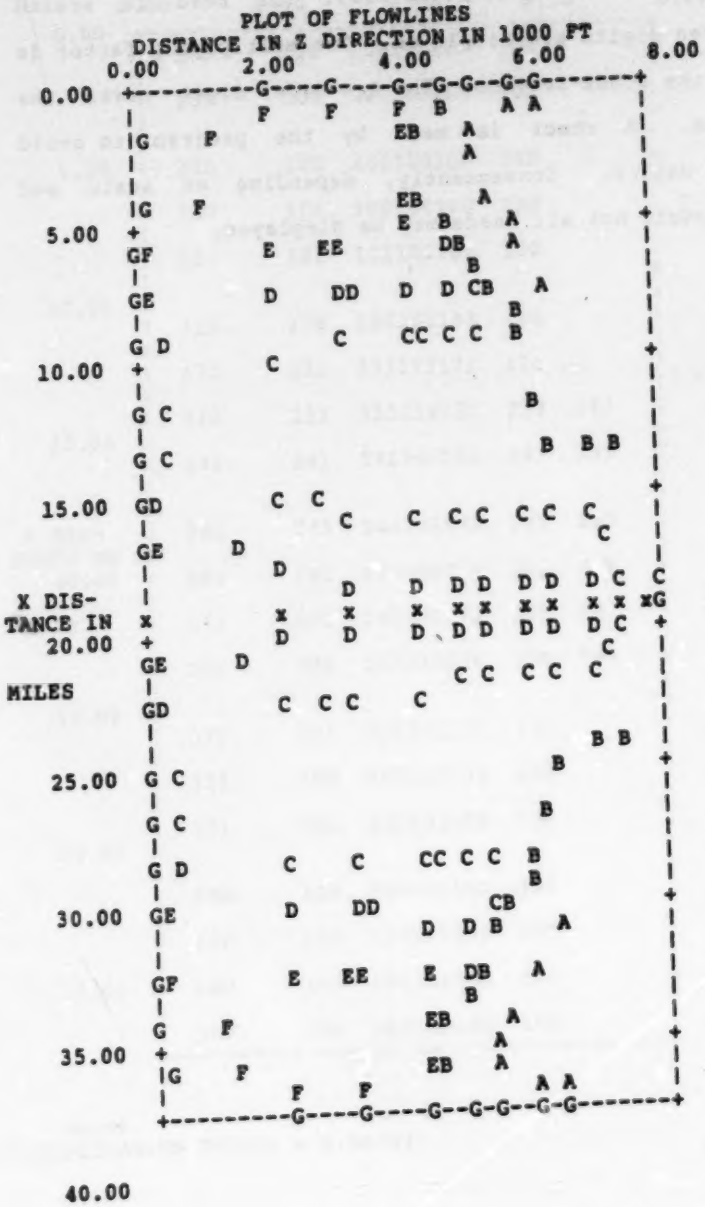
```
    |1.        1
    |   0   0   0   0   0   0   0   0 5-1 5-1 5-1 5-1   0   0   0   0   0   0   0   0
    |   0   0   0   0   0   0   0 5-1 5-1 5-1 5-1 5-1 5-1   0   0   0   0   0   0 1+4
    | 1+4   0   0   0   0   0 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4   0   0   0   0 1+1 1+1
    | 1+1 1+1 1+1 1+1 1+1 1+1 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1+1 1+1 1+1 1+1 1+1 1+1
P   | 1+1 1+1 1+1 1+1 1+1 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1+1 1+1 1+1 1+1 1+1
    | 1+1 1+1 1+1 1+1 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1+1 1+1 1+1
    | 1+1 1+1 1+1 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4
    | 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4
    | 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4

    |1.        1
    |   0   0   0   0   0   0   0   0 .05 .05 .05 .05   0   0   0   0   0   0   0   0
    |   0   0   0   0   0   0   0 .05 .05 .05 .05 .05 .05   0   0   0   0   0   0   0
    |   0   0   0   0   0   0 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4   0   0   0   0   0   0
    | .30 .30 .30 .30 .30 .30 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 .30 .30 .30 .30 .30 .30
POR | .30 .30 .30 .30 .30 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 .30 .30 .30 .30
    | .30 .30 .30 .30 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 .30 .30 .30
    | .30 .30 .30 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4
    | 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4
    | 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4
    | 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4 1-4

    |OOOOOOOOBBBBOOOOOOOO
    |OOOOOOOBBBBBBOOOOOOO
    |2OOOOOGGGGGGGGOOOOO2
    |AAAAAAGGGGGGGGAAAAAA
SEC |AAAAAGGGGGGGGGGAAAAA
    |AAAAGGGGGGGGGGGGAAAA
    |AAAGGGGGGGGGGGGGGAAA
    |GGGGGGGGGGGGGGGGGGGG
    |GGGGGGGGGGGGGGGGGGGG
    |GGGGGGGGGGGGGGGGGGGG

    |5280.     1000.     5.        2.        MILES     1000 FT
```

Output for Cross Sections

For cross-section models (M=1) additional output is written to a disk file called "input-file-name.PRINT". The additional output is four cross-sections showing selected flow-lines, head, travel time, and lithology.

The first of these cross-sections shows selected flow paths through the cross-section. The cross-section is examined by the program to determine if it contains more than one flow system. The program does this by examining the velocity along the sides and bottom of the cross-section. If a reversal in direction of movement occurs at the side or base of the cross-section, a boundary between flow systems exists there. A flow line is traced backward from this point to the top of the model and it's position is indicated on the cross-section by "x". This marks the system boundary. Within each distinct flow system, seven flow paths are traced. These paths, denoted by A, B, C, D, E, F, and G, represent .5, .1, .01, .001, .0001, .00001, and .000001, respectively, of the flow entering the top of the flow system. Points on the flow paths are shown where paths cross block faces. Each path begins at the top of the model and ends on the face of the discharge block. Points on the flow paths may overlap on the plot and only the highest point in the sequence A, B, C, D, E, F, G, x will be shown on the cross-section.

Attachment 5, continued



The second cross-section shows computed head. The head is scaled and no more than three digits are displayed. The multiplying factor is printed at the end of the cross-section. The low order digit marks the location of the value. A check is made by the program to avoid overlapping displayed values. Consequently, depending on scale and configuration of the model, not all heads may be displayed.

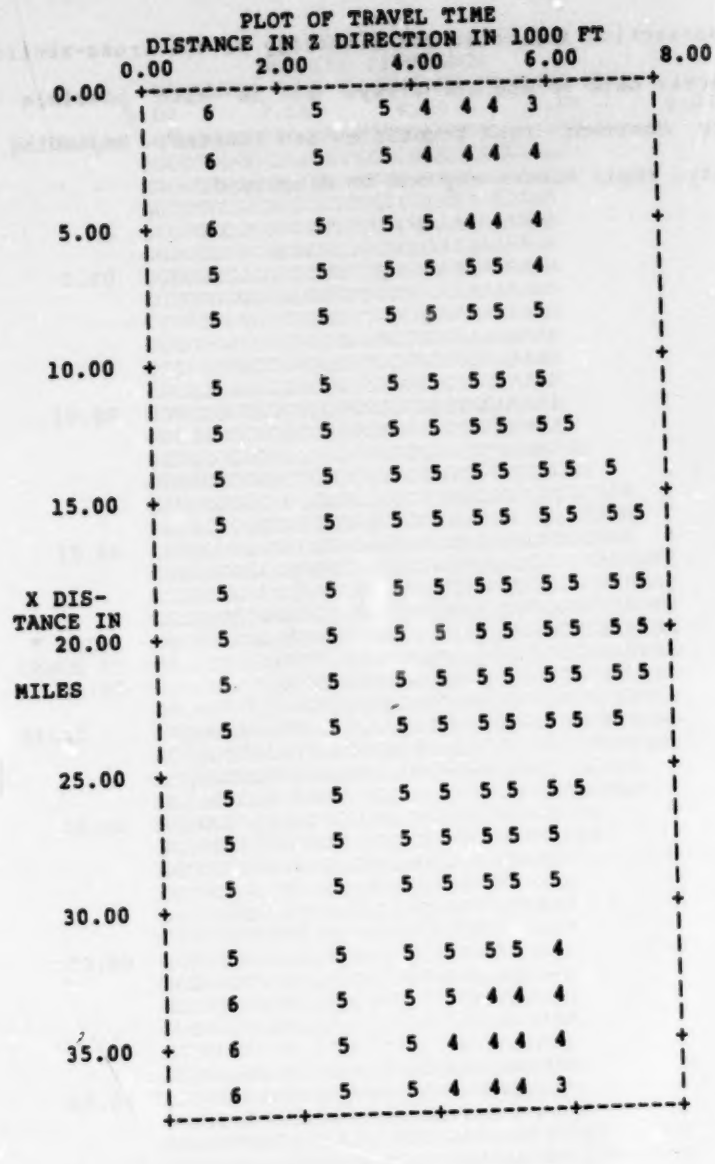
Attachment 5, continued

PLOT OF HEAD
DISTANCE IN Z DIRECTION IN 1000 FT

```
          0.00        2.00        4.00        6.00        8.00
     0.00 +----------+----------+----------+----------+
          |  100      100  100100100  100             |
          |  100      100  100100100  100             |
          |                                           |
     5.00 +  100      100  100100100  100             +
          |  100      100  100100100  100             |
          |  101      101  101100100  100             |
          |                                           |
    10.00 +                                           +
          |  111      109  107105103  100             |
          |  171      171  171171171  171             |
          |  230      231  233234236  239  243        |
    15.00 +                                           +
          |  241      241  241242242  243  243        |
          |                                           |
X DIS-    |  243      243  243243243  243  243        |
TANCE IN  |                                           |
    20.00 +  243      243  243243243  243  243        +
          |  241      241  241242242  243  243        |
MILES     |                                           |
          |  230      231  233234236  239  243        |
          |                                           |
    25.00 +                                           +
          |  171      171  171171171  171             |
          |  111      109  107105103  100             |
          |  101      101  101100100  100             |
    30.00 +                                           +
          |  100      100  100100100  100             |
          |  100      100  100100100  100             |
          |                                           |
    35.00 +  100      100  100100100  100             +
          |  100      100  100100100  100             |
          +----------+----------+----------+----------+
     40.00
MULTIPLICATION FACTOR = 1.0E+01
```

The third cross-section shows travel time in years. The displayed value is the exponent of the time. A display of "5" indicates that the time is greater or equal to 100,000 years and less than 1,000,000 years. The time is from the point to the discharge area. Travel time is computed for each active node. However, depending on scale and block dimensions, every time may not be displayed.

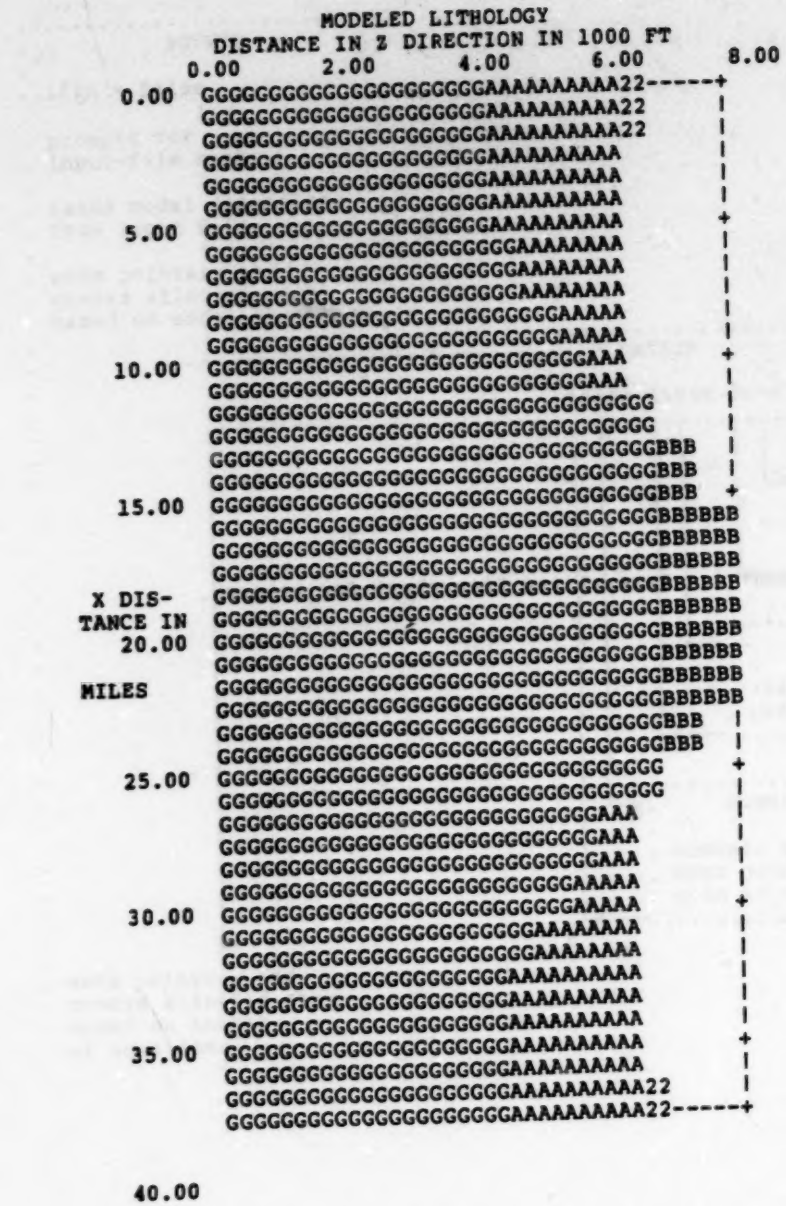
Attachment 5, continued



EXPONENT, 5 INDICATES 10.**5 <= VALUE < 10.**6

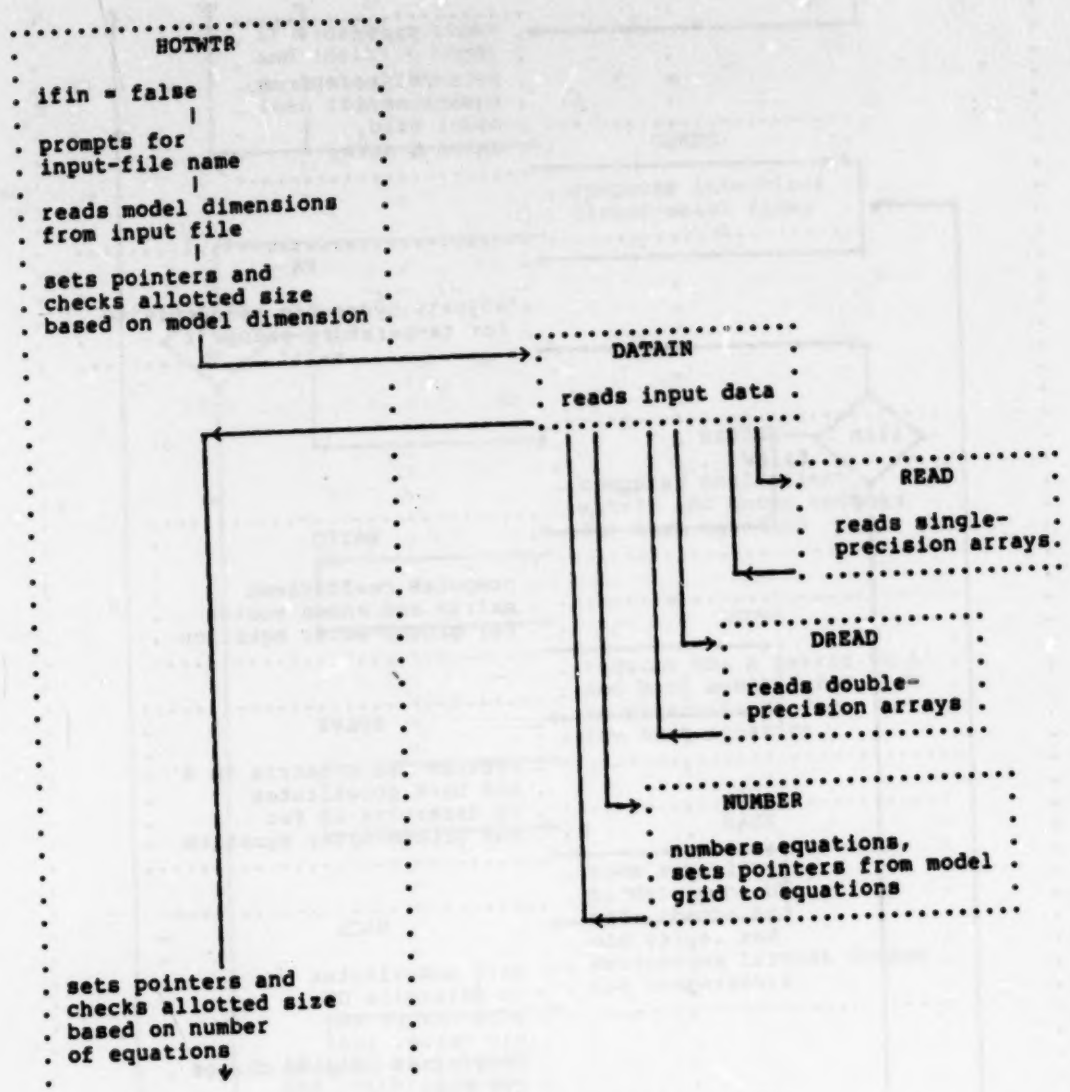
The last cross-section indicates the lithology of the cross-section. It shows the character data in the SEC array. It is also possible to indicate where the constant head boundaries are located. Depending on scale and block size, small blocks may not be displayed.

Attachment 5, continued

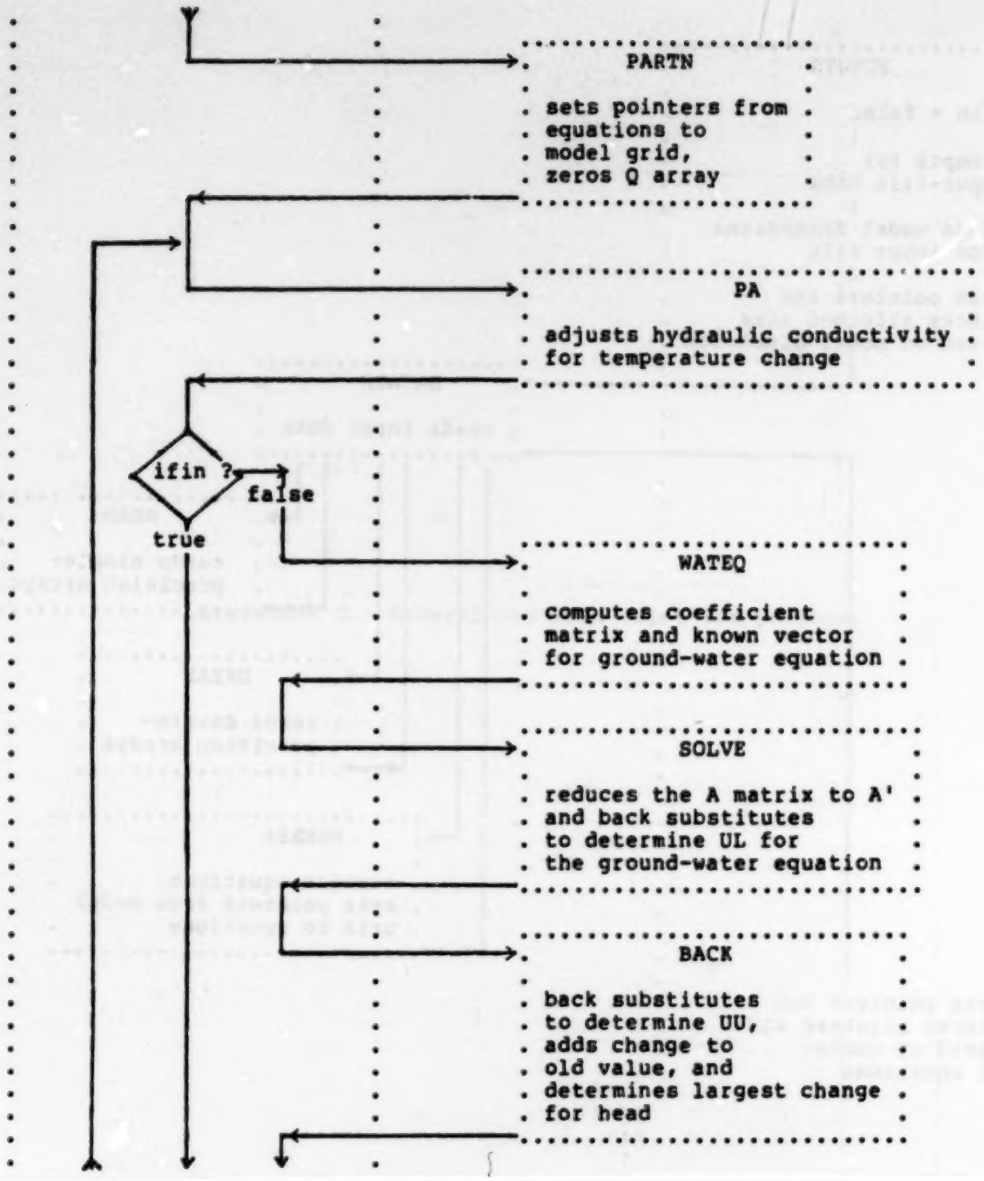


Attachment 6. Generalized flow chart for program.

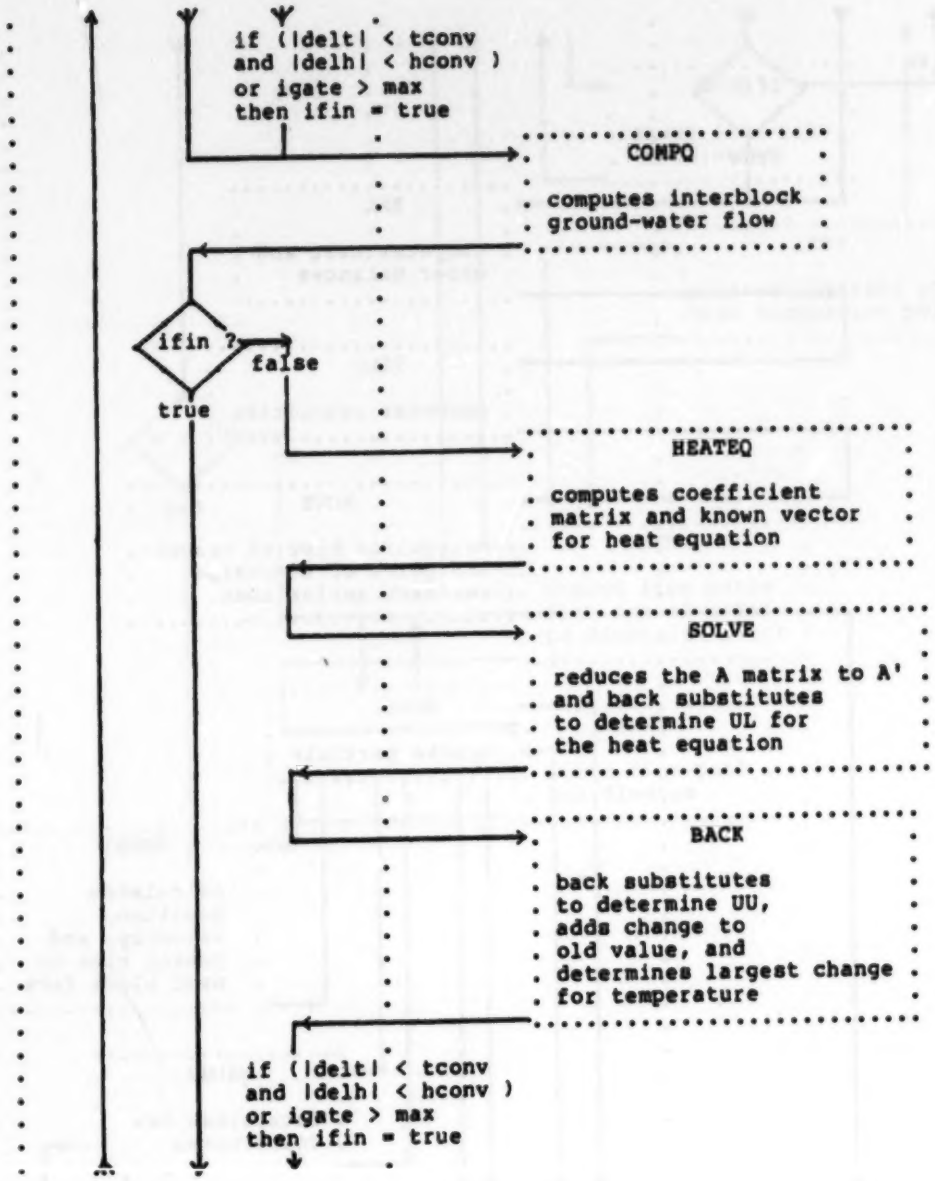
Attachment 6.


HOTWTR
ifin = false
prompts for input-file name
reads model dimensions from input file
sets pointers and checks allotted size based on model dimension
DATAIN
reads input data
READ
reads single-precision arrays.
DREAD
reads double-precision arrays
NUMBER
numbers equations, sets pointers from model grid to equations
sets pointers and checks allotted size based on number of equations

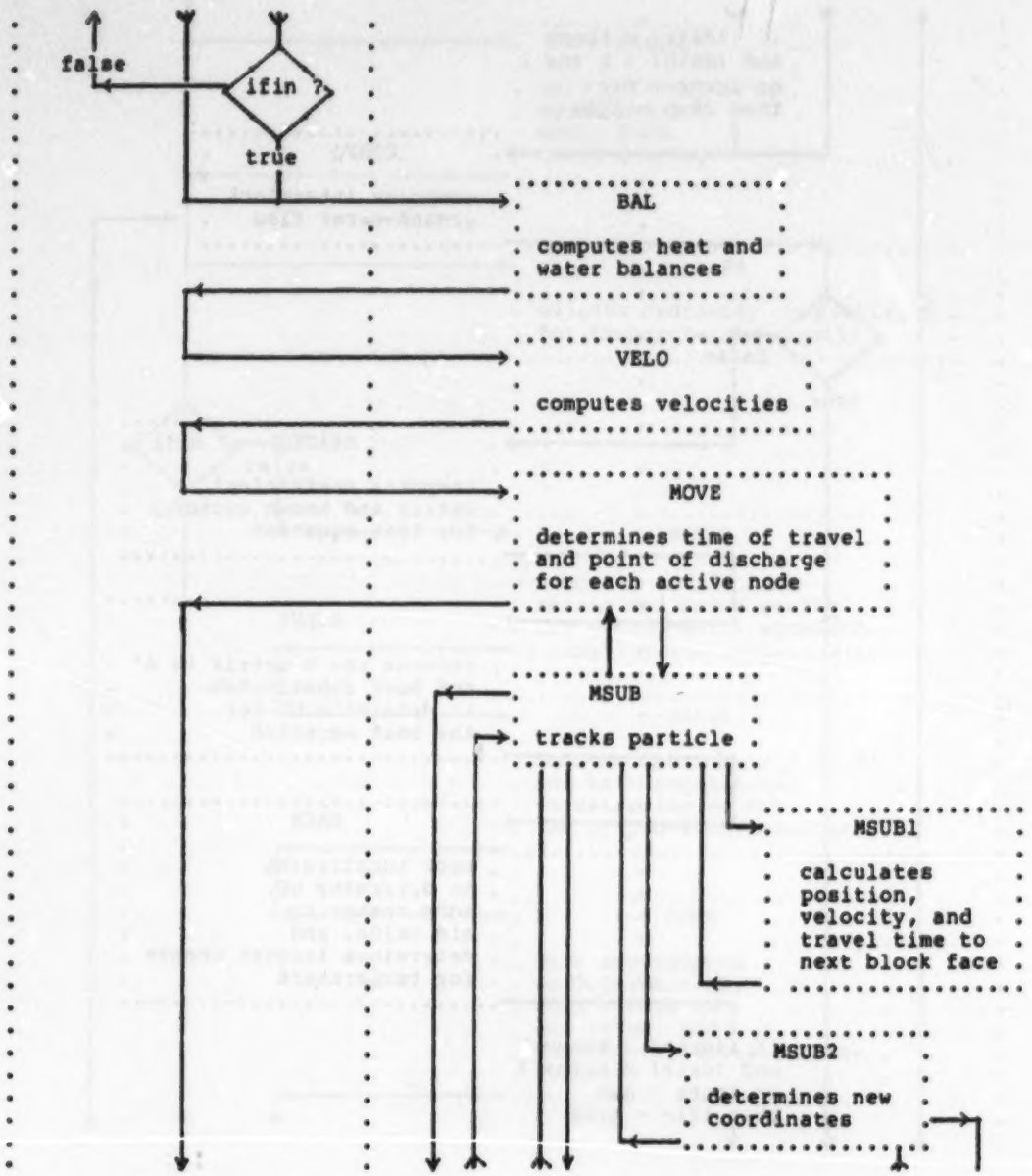
Attachment 6, continued


PARTN
sets pointers from equations to model grid, zeros Q array
PA
adjusts hydraulic conductivity for temperature change
ifin ?
false
true
WATEQ
computes coefficient matrix and known vector for ground-water equation
SOLVE
reduces the A matrix to A' and back substitutes to determine UL for the ground-water equation
BACK
back substitutes to determine UU, adds change to old value, and determines largest change for head

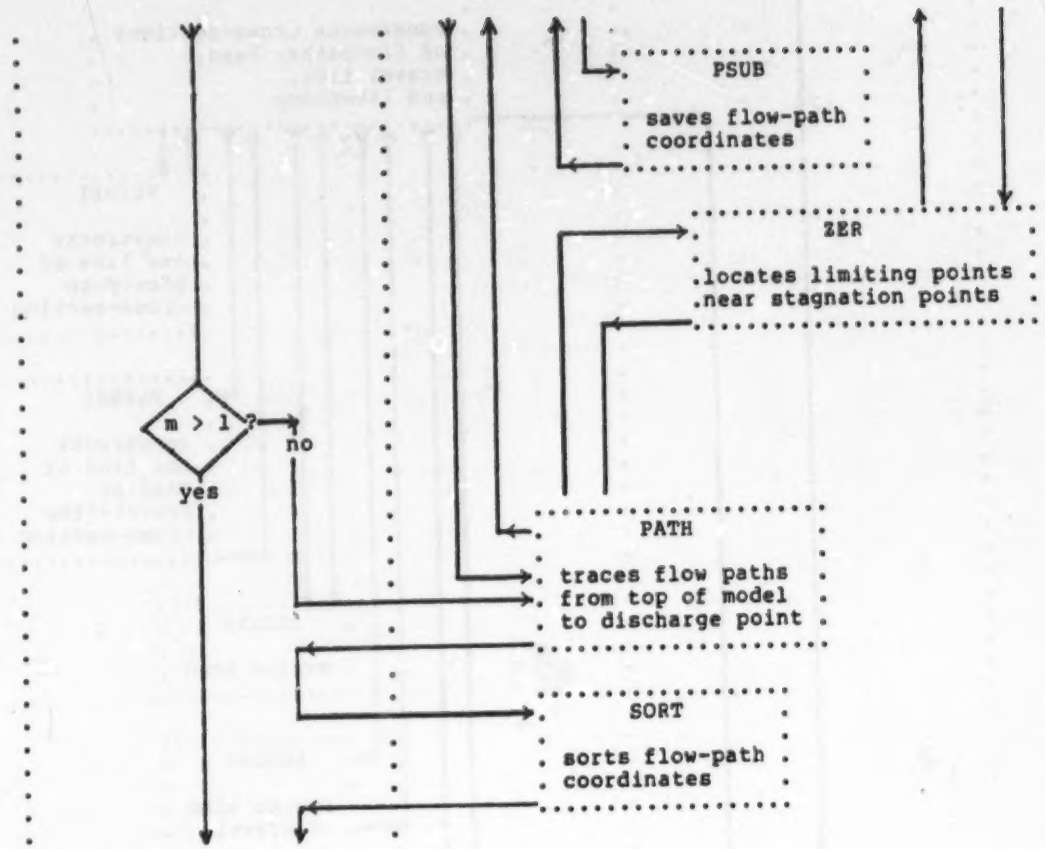
Attachment 6, continued



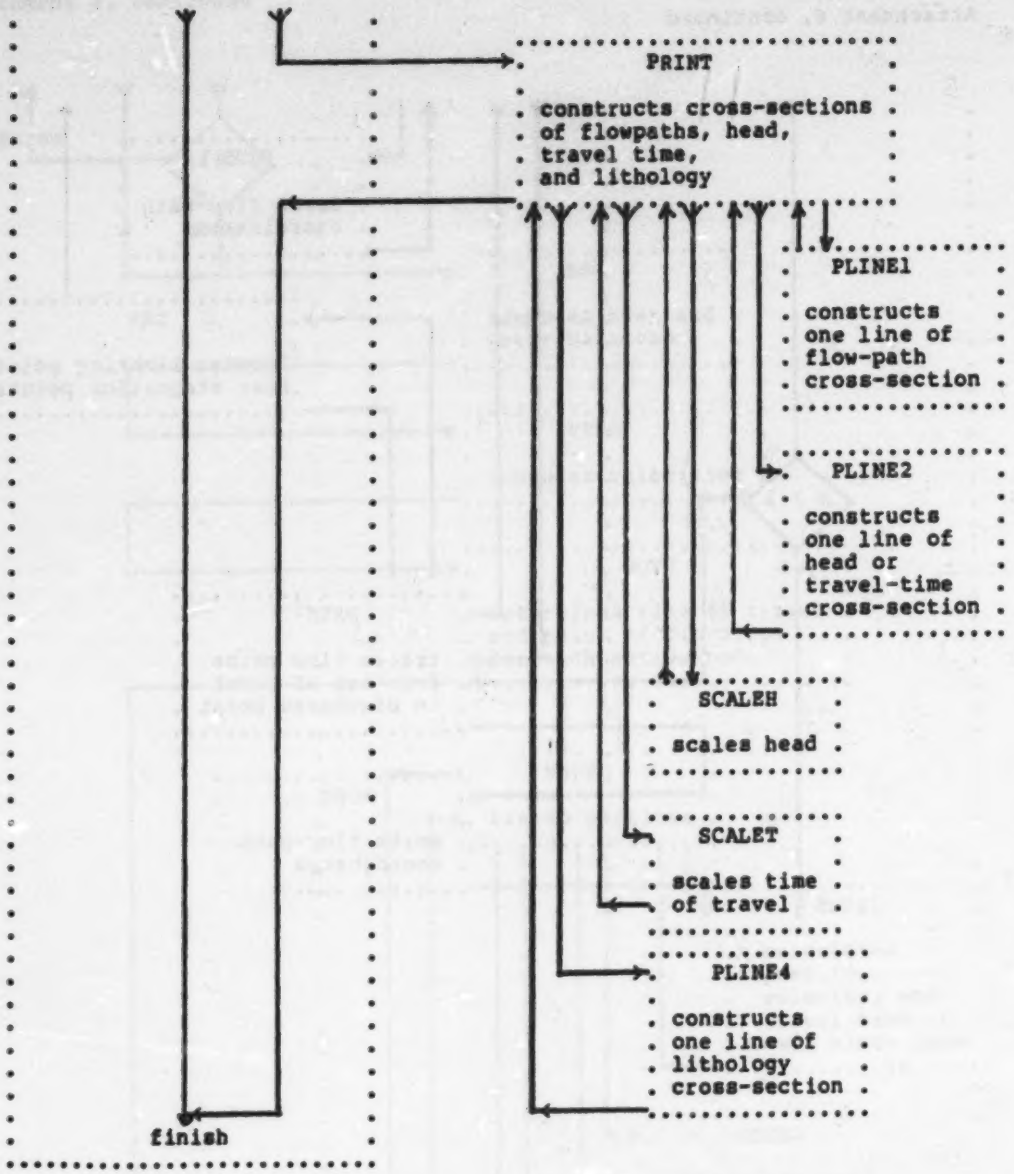
Attachment 6, continued



Attachment 6, continued



Attachment 6, continued



☆ U.S. GOVERNMENT PRINTING OFFICE:1985—676-554 / 45024